PROCEEDINGS OF THE FOURTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 4th '99)

Vol. 1

Jan. 19-Jan. 22, 1999
B-Con Plaza, Beppu, Oita, JAPAN

Editors : Masanori Sugisaka and Hiroshi Tanaka
ISBN4-9900462-9-3

Proceedings of The Fourth International Symposium on

ARTIFICIAL LIFE AND ROBOTICS

(AROB 4th '99)

Challenge for Complexity

January 19-22, 1999
B-Con Plaza, Beppu, Oita, JAPAN

Editors: Masanori Sugisaka and Hiroshi Tanaka

ISBN4-9900462-9-3, 1999

THE FOURTH INTERNATIONAL SYMPOSIUM
ON
ARTIFICIAL LIFE AND ROBOTICS
(AROB 4th '99)

ORGANIZED BY

Oita University under The Sponsorship of
Ministry of Education, Science, Sports, and Culture,
Japanese Government
Organizing Committee of Artificial Life and Robotics

CO-SPONSORED BY

Santa Fe Institute(SFI, USA)
The Institute of Electrical Engineers of Japan(IEEJ, Japan)
The Robotics Society of Japan(RSJ, Japan)
The Society of Instrument and Control Engineers(SICE, Japan)

CO-OPERATED BY

Japan Robot Association(JARA, Japan)
The Institute of Electrical and Electronics Engineers,
Tokyo Section(IEEE, USA)
The Institute of Electronics, Information and
Communication Engineers(IEICE, Japan)
The Institute of System, Control and Information
Engineers(ISCIE, Japan)

SUPPORTED BY

Asahi Shimbun Publishing Company Oita Bureau
Beppu Municipal Government
Jiji Press
Kyodo News
Kyushu Bureau of International Trade and Industry, MITI
NHK Oita Station
Nihon Keizai Shimbun. Inc.
Nikkan Kogyo Shimbun
Nisinippon Simbun
OBS Broadcast Company
Oita Asahi Broadcasting
Oita Industrial Group Society
Oita Municipal Government
Oita System Control Society
Oita Prefectural Government
Oitagodo Shinbunsya
Science and Technology Agency
TOS Broadcast Company
The Mainichi Newspapers
The Yomiuri Shimbun

HONORARY PRESIDENT

M. Hiramatsu (Governor, Oita Prefecture)

HONORARY ADVISER

Y. Fujita (President, The Japan Academy)

ADVISORY COMMITTEE CHAIRMAN

M. Ito (Director, RIKEN)

ORGANIZING COMMITTEE

K. Abe (Tohoku University, Japan)
K. Akizuki (Waseda University, Japan)
S. Arimoto (Ritsumeikan University, Japan)
W. B. Arthur (Santa Fe Institute, USA)
C. Barrett (Los Alamos National Laboratory, USA)
Z. Bubunicki (Wroclaw University of Technology, Poland)
J. L. Casti (Santa Fe Institute, USA)
T. Christaller (GMD-German National Research Center for Information Technology, Germany)
J. M.Epstein (Brookings Institution, USA)
T. Fujii (RIKEN, Japan)
S. Fujimura (The University of Tokyo, Japan)
Y. Fujita (The Japan Academy, Japan)
T. Fukuda (Nagoya University, Japan)
M. Gen (Ashikaga Institute of Technology, Japan)
T. Gomi (AAI, Canada)
H. Hagiwara (Kyoto School of Computer Science, Japan)
I. Harvey (University of Sussex, UK)
P. Husbands (University of Sussex, UK)
M. Ito (RIKEN, Japan)
D. J. G. James (Coventry University, UK)
T. Jinzenji (Sanyoudenki Co. Ltd., Japan)
J. Johnson (The Open University, UK)
Y. Kakazu (Hokkaido University, Japan)
R. E. Kalaba (University of Southern California, USA)
H. Kashiwagi (Kumamoto University, Japan)
O. Katai (Kyoto University, Japan)
S. Kauffman (Santa Fe Institute, USA)
J. H. Kim (KAIST, Korea)

H. Kimura (The University of Tokyo, Japan)
S. Kitamura (Kobe University, Japan)
H. Kitano (Sony computer Science Laboratory Inc., Japan)
S. Kumagai (Osaka University, Japan)
K. Kyuma (Mitsubishi Electric Co., Japan)
C. G. Langton (Santa Fe Institute, USA)
J. J. Lee (KAIST, Korea)
C. Looney (University of Nevada-Reno, USA)
G. I. Marchuk (Russian Academy of Science, Russia)
G. Matsumoto (RIKEN, Japan)
K. Matsuno (MITI, AIST, Japan)
H. Miura (The University of Tokyo, Japan)
H. Mushya (The University of Tokyo, Japan)
T. Nagata (Inst. of Sys. and Inf. Tech. / Kyushu, Japan)
M. Nakamura (Saga University, Japan)
H. H. Natsuyama (Advanced Industrial Materials, USA)
Y. Nishikawa (Osaka Institute of Technology, Japan)
R. G. Palmer (Santa Fe Institute, USA)
R. Pfeifer (University of Zurich-Irchel, Switzerland)
M. Raibert (MIT, USA)
S. Rasmussen (Santa Fe Institute, USA)
T. S. Ray (Santa Fe Institute, USA)
P. Schuster (Santa Fe Institute, USA)
T. Shibata (MITI, MEL, Japan)
K. Shimohara (ATR, Japan)
C. Sommerer (ATR, Japan)
L. Steels (VUB AI Laboratory, Belgium)
M. Sugisaka (Oita University, Japan) (Chairman)
K. Tamura (Tyukyou University, Japan)
S. Tamura (Osaka University, Japan)
H. Tanaka (Tokyo Medical & Dental University, Japan)
Y. Tokura (NTT, Japan)
N. Tosa (ATR, Japan)
K. Tsuchiya (Kyoto University, Japan)
S. Ueno (Kyoto School of Computer Science, Japan)
A. P. Wang (Arizona State University, USA)
W. R. Wells (University of Nevada-Las Vegas, USA)
T. Yamakawa (Kyushu Institute of Technology, Japan)
Y. G. Zhang (Academia Sinica, China)

STEERING COMMITTEE

M. Asada (Osaka University, Japan)
H. Asama (RIKEN, Japan)
Z. Bubunicki (Wroclaw University of Technology, Poland)
J. L. Casti (Santa Fe Institute, USA) (Co-chairman)

S. Fujimura (The University of Tokyo, Japan)
T. Fukuda (Nagoya University, Japan)
D. J. G. James (Coventry University, UK)
H. Kashiwagi (Kumamoto University, Japan)
H. Kimura (The University of Tokyo, Japan)
J. J. Lee (KAIST, Korea)
G. Matsumoto (RIKEN, Japan)
M. Nakamura (Saga University, Japan)
T. S. Ray (ATR, Japan)
K. Shimohara (ATR, Japan)
M. Sugisaka (Oita University, Japan) (Chairman)
H. Tanaka (Tokyo Medical & Dental University, Japan)
K. Tsuchiya (Kyoto University, Japan)
K. Ueda (Kobe University, Japan)
S. Ueno (Kyoto School of Computer Science, Japan)
W. R. Wells (University of Nevada-Las Vegas, USA)
Y. G. Zhang (Academia Sinica, China)

PROGRAM COMMITTEE

K. Aihara (The University of Tokyo, Japan) (Co-chairman)
S. Arimoto (Ritsumeikan University, Japan)
W. Banzhaf (University of Dortmund, Germany)
J. L. Casti (Santa Fe Institute, USA)
T. Fujii (RIKEN, Japan)
S. Fujimura (The University of Tokyo, Japan)
T. Fukuda (Nagoya University, Japan)
M. Gen (Ashikaga Institute of Technology, Japan)
I. Harvey (University of Sussex, UK)
T. Hasegawa (Kyushu University, Japan)
H. Hashimoto (The University of Tokyo, Japan) (Co- chairman)
K. Hirasawa (Kyushu University, Japan)
P. Husbands (University of Sussex, UK)
T. Ishimatsu (Nagasaki University, Japan)
Y. Kakazu (Hokkaidou University, Japan)
S. Kawaji (Kumamoto University, Japan)
S. Kawata (Tokyo Metropolitan University, Japan)
J. H. Kim (KAIST, Korea)
S. Kitamura (Kobe University, Japan)
H. Kitano (Sony computer Science Laboratory Inc., Japan)
T. Kitazoe (Miyazaki University, Japan)
H. Kobatake (Tokyo University of Agriculture & Technology, Japan)
K. Kosuge (Tohoku University, Japan)
K. Kumamaru (Kyushu Institute of Technology, Japan)
J. J. Lee (KAIST, Korea)
T. Nagata (Inst. of Sys. and Inf. Tech. / Kyushu, Japan)

M. Nakamura (Saga University, Japan)
K. Nakano (Fukuoka Institute of Technology, Japan)
Y. Nishikawa (Osaka Institute of Technology, Japan)
T. Omori (Tokyo University of Agriculture & Technology, Japan)
K. Okazaki (Fukui University, Japan)
R. Pfeifer (University of Zurich-Irchel, Switzerland)
T. S. Ray (ATR, Japan) (Co- chairman)
T. Sawaragi (Kyoto University, Japan)
T. Shibata (MITI, MEL, Japan)
K. Shimohara (ATR, Japan)
S. Shin (The University of Tokyo, Japan)
M. Sugisaka (Oita University, Japan)
H. Tamaki (Kobe University, Japan)
H. Tanaka (Tokyo Medical & Dental University, Japan)(Chairman)
N. Tosa (ATR, Japan)
K. Tsuchiya (Kyoto University, Japan)
Y. Uchikawa (Nagoya University, Japan)
K. Ueda (Kobe University, Japan)
S. Ueno (Kyoto School of Computer Science, Japan)
M. Wada (Hokkaido University, Japan)
X. Wang (Oita University, Japan)
K. Watanabe (Saga University, Japan)
T. Yamakawa (Kyushu Institute of Technology, Japan)
X. Yao (The University of New South Wales, Australia)

LOCAL ARRANGEMENT COMMITTEE

T. Ezaki (Director Research and Development Center,
Oita University, Japan)
Y. Fujita (Oita University, Japan)
T. Hano (Oita University, Japan)
H. Ikeuchi (Oita University, Japan)
T. Ito (Oita University, Japan)
K. Kinosita (Oita University, Japan)
A. Kuriyakawa (Oita University, Japan)
E. Kusayanagi (Oita University, Japan)
T. Matsuo (Oita University, Japan)
Y. Morita (Oita University, Japan)
T. Nabeshima (Oita University, Japan)
K. Ohta (Oita-Aist Joint Research Center, Japan)
M. Sugisaka (Oita University, Japan)
A. Tominaga (Oita University, Japan)
K. Yoshida (Oita University, Japan)

TOPICS

Hardware based topics are welcome
in the fields given by

Artificial Brain Research
Artificial Intelligence
Artificial Life
Artificial Living
Artificial Mind Research
Brain Science
Chaos
Cognitive Science
Complexity
Computer Graphics
Evolutionary Computations
Fuzzy Control
Genetic Algorithms
Human-Machine Cooperative Systems
Innovative Computations
Intelligent Control and Modeling
Micromachines
Micro-Robot World Cup Soccer Tournament
Mobile Vehicles
Neural Networks
Neurocomputers
Neurocomputing Technologies and Their Applications for Hardware
Robotics
Robust Virtual Engineering
Virtual Reality
Related Fields

COPYRIGHT

Accepted papers will be published in the Proc. of AROB and some of high quality papers in the Proc. will be requested to re-submit for the consideration of publication in a new international journal ARTIFICIAL LIFE AND ROBOTICS (Springer) and APPLIED MATHEMATICS AND COMPUTATION (North-Holland).

All correspondence related to the symposium should be addressed to :

AROB Secretariat
c/o Sugisaka Laboratory
Dept. of Electrical and Electronic Engineering, Oita University
700 Dannoharu, Oita 870-1192, JAPAN
TEL +81-97-554-7831
FAX +81-97-554-7841
E-MAIL arob@cc.oita-u.ac.jp

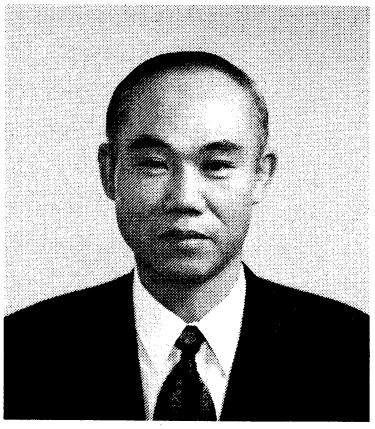
WWW Home Page http://arob.cc.oita-u.ac.jp/

AROB 4th '99 is supported financially by the following Companies

Aioi Seiki Inc.
All Nippon Airways Co., Ltd.
Hiji High-Tech Co., Ltd.
Isahaya Electronics Corporation
Ishii Tool & Engineering Corporation
Japan Air System
Japan Airlines
Nishinippon Plant Engineering and Construction Co., Ltd.
Oita Gas Co., Ltd.
Sanwa Shurui Co., Ltd.
Shintsurukai Kosan Co., Ltd.
Yanai Denki Kogyo Co., Ltd.
Yatsushika Sake-Brewing Co., Ltd.

PREFACE

Masanori Sugisaka
General Chairman of AROB
(Professor, Oita University)



It is my great honor to invite you all to The Fourth International Symposium on Artificial Life and Robotics(AROB 4th '99), organized by Oita University under the sponsorship of Ministry of Education, Science, Sports, and Culture(Monbusho), Japanese Government and co-sponsored by Santa Fe Institute(SFI), USA, SICE, RSJ, and IEEJ, Japan. This symposium invites you all to discuss development of new technologies concerning Artificial Life and Robotics based on simulation and hardware in twenty first century. It is also our great honor to welcome active scientists and engineers as new members in our symposium from this year.

Since the first symposium was held in Beppu in 1996, the progress of researches on artificial life, complexity, and robotics has been expected in industries, business, etc. to contribute for human society. The special topic in AROB 4th '99 is the challenge for complexity.

This symposium is also financially supported by not only Monbusho but also other private companies. I would like to express my sincere thanks to Monbusho, private companies, and all people who contributed to this symposium.

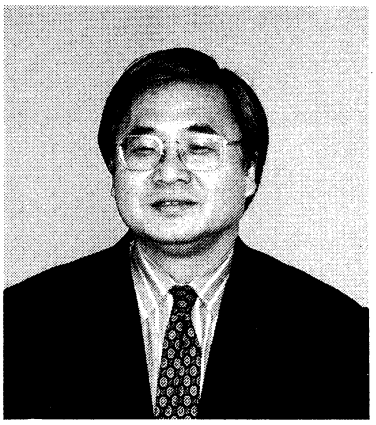
We hope that AROB 4th '99 will become a celebration to the establishment of our international joint research institute on artificial life, complexity and robotics for twenty first century by the support of Monbusho's program of center of excellence. I hope that you will obtain fruitful results by exchanging ideas through discussions during the symposium and also will enjoy your stay in Beppu, Oita.

I am looking forward to meeting you in Beppu.

Masanori Sugisaka
M. Sugisaka

January 12, 1999

PREFACE



Hiroshi Tanaka
Program Chairman of AROB
(Professor, Tokyo Medical and Dental University)

On behalf of the program committee, it is truly my great honor to invite you all to The Fourth International Symposium on Artificial Life and Robotics (AROB 4th '99). This symposium is made possible owing to the cooperation of Oita University and Santa Fe Institute. We are also debt to Japanese academic associations such as SICE, RSJ, IEEJ and several private companies. I would like to express my sincere thanks to all of those who make this symposium possible.

As is needless to say, the complex systems approach now attracts wide interests as a new paradigm of science and engineering not only in the traditional natural sciences fields like life science, comuter science and robotics but also in more social fields such as linguistics, ecology, sociology and economy. This shows that complex systems approach is now eagerly expected to become one of universal methodology of science to resolve many grand challenges that remain unsolved throughout this century. We hope this symposium becomes a forum for exchange of the ideas of the attendants from various fields who are interested in the future possibility of complex systems approach.

I am looking forward to meeting you in Beppu.

Hiroshi Tanaka

H. Tanaka

January 12, 1999

TECHNICAL PAPER INDEX

Plenary and Open Symposium Lectures (Invited Talks)

P1-1 *Computer simulation of gene flow in the malaria vector, anopheles gambiae, in West Africa* ……OP-1

C. Taylor (University of California-Los Angels, USA)

P1-2 *Discussion of "living" machines using chaotic control* ……OP-2

M. W. Tilden (Los Alamos National Laboratory, USA)

P1-3 *Complexity and Control* ……OP-4

H. Kimura (The University of Tokyo, Japan)

OS-1 *Origins of life : from recent topics* ……OP-5

T. Oshima (Tokyo University of Pharmacy and Life Science, Japan)

OS-2 *The mind of the brain* ……OP-7

G. Matsumoto (RIKEN, Japan)

OS-3 *BizSim: The world of business in a abox* ……OP-15

J. L. Casti (Santa Fe Institute, USA)

P2 *A universal origin of life* ……OP-21

S. Rasmussen, S. Colgate, K. Lachner, B. Lehnert, J. Solem, D. Whitten
Los Alamos National Laboratory, USA)
S. Rasmussen, S. Colgate (Santa Fe Institute, USA)
L. Luisi (ETHZ Zurich, Switzerland)
B. Mayer (University of Vienna, Austria)

P3 *Evolving robot morphology and control* ……OP-22

R. K. Belew, C. Mautner (University of California-San Diego, USA)

P4 *Grasping impact force control of a flexible robotic gripper using piezoelectric actuator* ……OP-28

W. R. Wells, W. Yim (University of Nevada-Las Vegas, USA)

P5 *Learning algorithms in a class of knowledge-based systems* ……*OP-34*
Z. Bubnicki (Wroclow University of Technology, Poland)

Invited Lectures (Invited Talks)

IT1-1 *Evolution of differentiated multi-threaded digital organisms* ……*I-1*
T. Ray, J. Hart (ATR, Japan)

IT1-2 *Robot populations and their controlled evolution* ……*I-2*
D. J. G. James (Coventry University, UK)
V. G. Rumchev (Curtin University of Technology, Australia)

IT1-3 *Ola: what goes up, must fall down* ……*I-9*
H. H. Lund, J. A. Arendt, J. Fredslund, L. Pagliarini (University of Aarhus, Denmark)

IT2-1 *Dynamic bidirectional associative memory using chaotic neurons* ……*I-16*
J. J. Lee (KAIST, Korea)

IT2-2 *A bottom-up way to develop multi-cellular digital organisms* ……*I-22*
Y. G. Zhang, X. Wu (Academia Sinica, China)
M. Sugisaka (Oita University, Japan)

IT3-1 *Human-machine cooperative system for natural language processing* ……*I-27*
R. Dai, Y. Fei (Chinese Academy of Science, China)

IT3-2 *An adaptive classifier system tree for extending genetic based machine learning in dynamic environment* ……*I-31*
D. Hu, R. Jiang, Y. Luo (Tsinghua University, China)

IT4-1 *Winning strategies for robot war games* ……*I-35*
B. Stilman (University of Colorado at Denver, USA)

IT4-2 *Insect size robots with insect level intelligence* ……*I-41*
J. D. Nicoud (Swiss Federal Institute of Technology, Switzerland)

S-A1: Theater, movie and A-life (Invited Session) (Room A)

A1-1 *Theater, movie with A-life -Romeo & Juliet in hades as A-life based cinema-* 1
N. Tosa (ATR, Japan)

A1-2 *Treatment of nonverbal behaviors for A life-based computer actors* 5
R. Nakatsu, A. Solomides, N. Tosa (ATR, Japan)

A1-3 *Technoetic theatre: performance and enactment in the dramaturgy of artificial life* 12
R. Ascott (University of Plymouth and Wales College, UK)

A1-4 *Sociality of robots in symbiotic relations with humans* 16
T. Ono, M. Imai, T. Etani (ATR, Japan)

S-B1: Intelligent control and robotics I (Invited Session) (Room B)

B1-1 *Creation of subjective value through physical interaction between human and machine* 20
T. Shibata, K. Tanie (Ministry of International Trade and Industry, Japan)

B1-2 *Task sharing in a multiple robot system controlled by macro-commands of an operator* 24
K. Ohkawa (University of Tsukuba, Japan)
T. Shibata, K. Tanie (Ministry of International Trade and Industry, Japan)

B1-3 *Evolutionary parallel collaborative computation on intelligent agents* 28
T. Yamaguchi, N. Kohata, M. Takahide, T. Baba (Utsunomiya University, Japan)
H. Hashimoto (The University of Tokyo, Japan)

B1-4 *Development of an intelligent data carrier (IDC) system and its applications* 34
D. Kurabayashi, H. Asama, T. Fujii, H. Kaetsu, I. Endo (RIKEN, Japan)

B1-5 *Physical agent and media on computer network* 40
Y. Kunii (Chuo University, Japan)
H. Hashimoto (The University of Tokyo, Japan)

B1-6 *Fast estimation of motion parameters for vehicle-camera using focus of expansion* ……46
Z. Hu, K. Uchimura, S. Kawaji (Kumamoto University, Japan)

B1-7 *Intelligent control of robots in coordination* ……50
K. Kosuge, Y. Hirata, K. Takeo (Tohoku University, Japan)
H. Asama, H. Kaetsu, K. Kawabata (RIKEN, Japan)

S-B2: Complexity (General Session) (Room B)

B2-1 *A study on coalition formation process in the iterated multiple lake game* ……54
T. Yamashita, K. Suzuki, A. Ohuchi (Hokkaido University, Japan)

B2-2 *Estimating an independent component based on the mutual information against the residual component* ……58
T. Iwamoto (Mitsubishi Electric Corp., Japan)

B2-3 *Rotational model of the stock prices(I)* ……61
S. Maekawa, Y. Fujiwara (Communications Research Laboratory, Japan)

B2-4 *Rotational model of the stock prices(II)* ……65
Y. Fujiwara, S. Maekawa (Communications Research Laboratory, Japan)

B2-5 *Multi agent approach against computer virus: An immunity-based system* ……69
T. Okamoto (Nara Institute of Science and Technoloty, Japan)
Y. Ishida (Toyohashi University of Technology, Japan)

B2-6 *"Life Spacies": a genetic text-to-form editor on the internet* ……73
C. Sommerer, L. Mignonneau (ATR, Japan)

S-B3: Chaos engineering (Invited Session) (Room B)

B3-1 *Learning in software driven neural networks with temporal coding and functional connectivity* ……78
M. Watanabe, K. Aihara (The University of Tokyo, Japan)

B3-2 *Retrieval characteristics of associative chaotic neural networks with weighted pattern storage* ……82
M. Adachi (Tokyo Denki University, Japan)
K. Aihara (The University of Tokyo, Japan)

B3-3 *Chaotic evolution in a game of a host and two parasites* ……86
G. Hori (RIKEN, Japan)
K. Aihara (The University of Tokyo, Japan)

B3-4 *IC implementation of a multi-internal-state chaotic neuron model with unipolar and bipolar output functions* ……90
Y. Horio, I. Kobayashi, H. Hayashi (Tokyo Denki University, Japan)
K. Aihara (The University of Tokyo, Japan)

B3-5 *Detecting nonlinear causality via nonlinear modeling* ……94
T. Ikeguchi (Science University of Tokyo, Japan)

B3-6 *Short-term prediction about complex sequences in a blast furnace by bell-shaped radial basis function networks* ……98
T. Miyano, H. Shibuta (Sumitomo Metal Industries, Ltd., Japan)
K. Aihara (The University of Tokyo, Japan)

B3-7 *Some examples on reconstructing nonlinear dynamics by the generalized exponential autoregressive model* ……102
Z. Shi, Y. Tamura, T. Ozaki (The Institute of Statistical Mathematics, Japan)

S-C1: Intelligent control system-discrete event systems approach (Invited Session) (Room C)

C1-1 *Autonomous distributed control by multi agent nets* ……106
H. Nakajima, T. Miyamoto, S. Kumagai (Osaka University, Japan)

C1-2 *Estimate based limited lookahead control of discrete event systems with model uncertainty* ……110
S. Takai (Wakayama University, Japan)

C1-3 *Conflict resolution in continuous petri nets using linear programming* ……114
A. Tanaka, T. Ushio (Osaka University, Japan)
S. Kodama (Kinki University, Japan)

C1-4 *Performance improvement in an internal model control of discrete-event systems based on max-algebra* ……118
S. Masuda, A. Inoue, Y. Hirashima (Okayama University, Japan)
H. Suzuki (Toshiba Co., Ltd., Japan)

C1-5 *Autonomous distributed control of manufacturing system by scene transition nets* ……122
S. Kawata (Tokyo Metropolitan University, Japan)
M. Haruyama (Mitsubishi Heavy Industries, Ltd., Japan)

C1-6 *Introduction of biochemical allosteric property for creating a new electrical signal transmission system under the time minimum optimization* ……126
H. Hirayama (Asahikawa Medical College, Japan)
Y. Okita (National Institute of Special Education, Japan)

S-C2: Neural network and associative memory (Invited Session) (Room C)

C2-1 *Diffusively coupled multi-neural networks for optimization problems* ……130
R. Horie, E. Aiyoshi (Keio University, Japan)

C2-2 *Learning algorithm for intermodular connection of multimodular associative networks* ……134
M. Mochizuki, H. Minamitani (Keio University, Japan)

C2-3 *Multi-winners self-organizing multidirectional associative memory* ……138
J. Huang, M. Hagiwara (Keio University, Japan)

C2-4 *Bill money recognition by using the LVQ method* ……142
T. Kosaka (Glory Inc., Japan)
S. Omatsu (Osaka Prefecture University, Japan)

C2-5 *Neuro-approach for hard disk driver position control* ……146

T. Fujinaka, M. Yoshioka, S. Omatsu (Osaka Prefecture University, Japan)

C2-6 *A saccadic model with distributed feedback mechanism: simulations of normal and interrupted saccades* ……150
K. Arai (Mitsubishi Chemical Corp., Japan)
S. Das (ITT Systems & Sciences Corp., Japan)
E. L. Keller (University of California-Berkeley, USA)
E. Aiyoshi (Keio University, Japan)

S-C3: Emergent system design I (Invited Session) (Room C)

C3-1 *Reinforcement learning approach to cooperative carrying problem* ……154
K. Kawakami, K. Ohkura, K. Ueda (Kobe University, Japan)

C3-2 *On the emergence of motion patterns in locomotion systems* ……158
M. M. Svinin, K. Yamada, K. Ueda (Kobe University, Japan)

C3-3 *Towards man-machine creativity in conceptual design* ……162
V. V. Kryssanov, H. Tamaki, S. Kitamura (Kobe University, Japan)

C3-4 *Self-organization process in interactive manufacturing environment* ……166
N. Fujii, I. Hatono, K. Ueda (Kobe University, Japan)

S-D1: Intelligent control and robotics II (Invited Session) (Room D)

D1-1 *A intelligent control of robot manipulator by visual feedback* ……170
S. H. Han (Kyungnam University, Korea)
J. I. Bae (Pukyung National University, Korea)
M. H. Lee (Pusan National University, Korea)

D1-2 *Robust predictive control of robot manipulators* ……175
M. C. Han, M. H. Lee (Pusan National University, Korea)

D1-3 *Implementation of virtual reality through the fusion of visual and force information* ……179
S. K. An, S. J. Han, J. M. Lee, M. H. Lee (Pusan National University, Korea)

D1-4 *Hardware Implementation of fuzzy logic controller for speed of a DC series motor using an adaptive evolutionary computation* ……183
G. H. Hwang, K. J. Mun, J. H. Park, M. H. Lee (Pusan National University, Korea)

D1-5 *Development of a 3D graphic simulation tool for a SCARA robot* ……188
D. Y. Lee, J. W. Choi, M. H. Lee, K. Son, M. C. Lee, J. M. Lee (Pusan National University, Korea)
S. H. Han (Kyungnam University, Korea)

D1-6 *A polynomial fuzzy neural network for modeling and control* ……192
S. Kim, M. H. Lee (Pusan National University, Korea)

S-D2: Virtual reality I (General Session) (Room D)

D2-1 *Development of horseback riding therapy simulator with VR technology* ……196
O. Sekine, Y. Shinomiya, R. Nakajima (Matsushita Electric Works, Ltd., Japan)
T. Kimura (Nippon Medical School, Japan)

D2-2 *A novel application of face image processing to wearable computer –augmentation of human* memory for faces and names ……200
H. Mizoguchi, T. Shigehara, Y. Goto, T. Mishima (Saitama University, Japan)

D2-3 *Virtual earphone to whisper in a person's ear remotely by utilizing visual tracking and speakers array* ……204
H. Mizoguchi, T. Shigehara, Y. Goto, M. Teshiba, T. Mishima
(Saitama University, Japan)

D2-4 *Virtual concierge: a talking door-phone to sepak up visitor's name* ……208
H. Mizoguchi, T. Shigehara, Y. Goto, T. Mishima (Saitama University, Japan)

D2-5 *Realizing virtual wireless microphone to pick up human voice remotely and clearly* ……212
H. Mizoguchi, T. Shigehara, M. Teshiba, T. Mishima (Saitama University, Japan)

S-D3: Robotic applications based on artificial intelligence (Invited Session) (Room D)

D3-1 *A neural network based artificial life model for navigation of multiple autonomous mobile robots in the dynamic environment* ……216
S. K. Min, H. Kang (Chung-Ang University, Korea)

D3-2 *Cellular automata based neural networks (CABANN) for optimal path planning* ……220
Y. G. Jo, H. Kang (Chung-Ang University, Korea)

D3-3 *Evolving cellular automata neural systems 2* ……224
D. W. Lee, K. B. Sim (Chung-Ang University, Korea)

D3-4 *Generalized asymmetrical BAM* ……228
T. D. Eom, J. J. Lee (KAIST, Korea)

D3-5 *Artificial immune system for realization of cooperative strategies and group behavior in collective autonomous mobile robots* ……232
D. W. Lee, H. B. Jun, K. B. Sim (Chung-Ang University, Korea)

D3-6 *Genetic programming-based A life techniques for evolving collective robotic intelligence* ……236
D. Y. Cho, B. T. Zhang (Seoul National University, Korea)

D3-7 *A digital artificial brain architecture for mobil autonomous robots* ……240
A-P Uribe, E. Sanchez (Swiss Federal Institute of Technology-Lausanne, Switzerland)

S-E1: Complexity in life system (Invited Session) (Room E)

E1-1 *Complexities in biosystem* ……244
H. Tanaka (Tokyo Medical and Dental University, Japan)

E1-2 *The synthesis of gene with block automaton* ……246
M. Kinoshita, M. Wada (Hokkaido University, Japan)

E1-3 *Statistical approach to genetic algorithm* ……250
Y. Fujiwara (Communications Research Laboratory, Japan)

E1-4 *Toward the realization of an evolving ecosystem on cellular automata* ……254
H. Sayama (The University of Tokyo, Japan)

E1-5 *Prediction of deviant genetic codes -Why they evolve-* ……258
T. Maeshiro (ATR, Japan)

E1-6 *On a correlation between the degree of halting property and the qualitative behavior of abstract chemical system* ……262
Y. Suzuki, H. Tanaka (Tokyo Medical and Dental University, Japan)

S-E2: Robotics I (General Session) (Room E)

E2-1 *Control of a humanoid robot using a multi-freedoms motion capture device* ……266
S. Kurono, Y. Miyamoto (Kyushu Sangyo University, Japan)
S. Aramaki (Fukuoka University, Japan)

E2-2 *Robot path search including obstacles by GA* ……270
H. Yamamoto (Wakayama University, Japan)

E2-3 *A robot control/learning scheme with task compatibility* ……274
Q. Guo (Beijing Institute of Technology, China)

E2-4 *Fine motion strategy using skill-based backprojection in consideration of uncertainty in control and sensing* ……275
A. Nakamura, T. Suehiro, H. Tsukune (Electrotechnical Laboratory, Japan)
T. Ogasawara (Nara Institute of Science and Technology, Japan)

E2-5 *Sliding mode controller for robot manipulators with predetermined transient response* ……279
K. B. Park, T. Tsuji (Kyushu Institute of Technology, Japan)
J. J. Lee (KAIST, Korea)

E2-6 *A novel application of legged mobile robot to human robot collaboration* ……283
H. Mizoguchi, Y. Goto, K. Hidai, T. Shigehara, T. Mishima (Saitama University, Japan)

E2-7 *Internal state acquisition for reinforcement learning agent by using radial basis function neural network* ……287
H. Murao, S. Kitamura (Kobe University, Japan)

S-E3: Robotics II (General Session) (Room E)

E3-1 *Real-time search for autonomous mobile robot using the framework of anytime algorithm* ……291
K. Fujisawa, T. Suzuki, S. Okuma (Nagoya University, Japan)
S. Hayakawa (Toyota Technological Institute, Japan)
T. Aoki (Nagoya Municipal Industry Research Institute, Japan)

E3-2 *Incremental evolution of CAM-brain to control a mobile robot* ……297
G. B. Song, S. B. Cho (Yonsei University, Korea)

E3-3 *Path planning for mobile robot using a genetic algorithm* ……301
S. Tamura, M. Takuno, T. Hatanaka, K. Uosaki (Tottori University, Japan)

E3-4 *An evolvable NAND-logic circuit applied to a real mobile robot khepera* ……305
Y. Wei, M. M. Islam, R. Odagiri, T. Asai, K. Murase (Fukui University, Japan)

E3-5 *Cooperation of real mobile robots using communication* ……309
M. M. Islam, Y. Wei, R. Odagiri, T. Asai, K. Murase (Fukui University, Japan)

E3-6 *Solving the equations of constrained motion in a lower LIMB model* ……313
C. Itiki (University of San Paulo, Brazil)
R. Kalaba (University of Southern California, USA)
H. Natsuyama (Seasons Associates, USA)

E3-7 *Estimation of muscule parameters of a lower LIMB model* ……317
C. Itiki (University of San Paulo, Brazil)
R. Kalaba (University of Southern California, USA)
H. Natsuyama (Seasons Associates, USA)

S-A2: Emergent system design II (Invited Session) (Room A)

A2-1 *A principle of design of an autonomous mobile robot* ……320
K. Tsuchiya, K. Tsujita (Kyoto University, Japan)

A2-2 *Autonomous robot control by a neural network with dynamic rearrangement function* ……324
T. Kondo, A. Ishiguro, Y. Uchikawa (Nagoya University, Japan)
P. Eggenberger (University of Zurich, Switzerland)

A2-3 *Maintenance of diversity by means of thermodynamical selection rules for genetic problem solving* ……330
H. Kita (Tokyo Institute of Technology, Japan)
N. Mori (Osaka Prefecture University, Japan)
Y. Nishikawa (Osaka Institute of Technology, Japan)

A2-4 *Protein folding by a hierarchical genetic algorithm* ……334
O. Takahashi, H. Kita, S. Kobayashi (Tokyo Institute of Technology, Japan)

S-B4: Artificial life (General Session) (Room B)

B4-1 *Artificial behavior of cell-like structure with polarized elements* ……340
T. Kohashi, T. Takayanagi, K. Suzuki, A. Ohuchi (Hokkaido University, Japan)

B4-2 *Origin and evolution of early peptide-synthesizing biomachines by means of hierarchical sociogenesis of intracellular primitive tRNA-riboorganisms* ……344
K. Ohnishi, S. Hokari (Niigata University, Japan)
H. Yanagawa (Mitsubishi Kasei Institute of Life Sciences, Japan)

B4-3 *Long-term increase of complexity and functional diversification by contingent mutations in computational algorithms* ……350
S. Ohashi, Y. Kakazu (Hokkaido University, Japan)
S. Yoshii (University of Liverpool, UK)

B4-4 *Team plays of soccer agents based on evolutionary dynamic formations* ……354
T. Murata, M. Yamamoto, K. Suzuki, A. Ohuchi (Hokkaido University, Japan)

B4-5 *An analysis of DNA-based computing process* ……358
T. Hirayama, T. Shiba, M. Yamamoto, K. Tsutsumi
S. Takiya, M. Munekata, K. Suzuki, A. Ohuchi (Hokkaido University, Japan)

B4-6 *Self-organized critical behaviors of fish schools and emergence of group intelligence* ……362
Y. Narita, K. Hattori, Y. Kashimori, T. Kambara, (University of Electro-Communications, Japan)

S-B5: Virtual reality Ⅱ (Invited Session) (Room B)

B5-1 *The degree of human visual attention in the visual search* ……363
H. Mizuhara, J. L. Wu (Yamaguchi University, Japan)
Y. Nishikawa (Osaka Institute of Technology, Japan)

B5-2 *Human interactive characteristic between binocular disparity and occlusion for depth perception* ……367
J. L. Wu, H. Yoshida (Yamaguchi University, Japan)

B5-3 *A following-type force display for the virtual catch ball system* ……371
K. Kimura, J. L. Wu, M. Kitazawa, Y. Sakai (Yamaguchi University, Japan)

B5-4 *A shape input system for three dimensional object in the virtual space* ……375
J. L. Wu, M. Kitazawa, H. Harada (Yamaguchi University, Japan)

B5-5 *Human characteristics of visual and tactual distance perception on the front parallel-plane for teleoperation systems* ……379
T. Miyake, J. L. Wu, X. Y. Lei (Yamaguchi University, Japan)

B5-6 *Human characteristics on visual and accelerative perception for virtual simulator* ……383
J. L. Wu, T. Kiyooka (Yamaguchi University, Japan)

B5-7 *Human visual and auditory characteristic in the temporal frequency domain* ……387
J. L. Wu, O. Nobuki (Yamaguchi University, Japan)

S-B6: Behavior and stability in human-machine cooperative systems (Invited Session) (Room B)

B6-1 *Immune algorithm with immune network and major histocompatibility complex* ……391
N. Toma, S. Endo, K. Yamada (University of the Ryukyus, Japan)

B6-2 *Application of competitive co-evolution algorithm to iterated prisoner's dilemma* ……395
M. Nerome, S. Endo, K. Yamada, H. Miyagi (University of the Ryukyus, Japan)

B6-3 *Vector lyapunov functions method in stability and control theories for logic-dynamical systems* ……399
V. M. Matrosov (Russian Academy of Science, Russia)

B6-4 *AHP coeffecients optimization technique based on GA* ……403
T. Toma, M. R. Asharif (University of the Ryukyus, Japan)

B6-5 *Study on cooperative agents through adaptive focal point* ……407
S. Yamauchi, K. Yamada, S. Endo, H. Miyagi (University of the Ryukyus, Japan)

B6-6 *On schemes for analysis of stability and asymptotical estimations in critical cases of stability theory* ……411
I. V. Matrossov (Moscow State University, Russia)

S-C4: Genetic algorithms I (General Session) (Room C)

C4-1 *Parallel distributed architectures of biologically inspired parameter-free genetic algorithm for simulating ecosystems* ……415
H. Sawai, S. Adachi (Kansai Advanced Research Center, Japan)
S. Kizu (Toshiba R & D Center, Japan)

C4-2 *A distributed system inspired from the immune system: an application to control* ……419
Y. Ishida (Toyohashi University of Technology, Japan)

C4-3 *Evolution of vision system for stereo perception by genetic algorithm* ……423
W. Nian, K. Okazaki (Fukui University, Japan)
S. Tamura (Osaka University, Japan)

C4-4 *Immune algorithm with adaptive memory* ……427
M. Yonezu, T. Yoshida, M. Nakanishi (Keio University, Japan)

S-C5: Intelligent mechatronics control (Invited Session) (Room C)

C5-1 *Neurocontroller for load swing suppression of a jib crane on a floating bed* ……431
F. Tabuchi, E. Uezato, H. Kinjo, T. Yamamoto (University of the Ryukyus, Japan)

C5-2 *Self-organization of emotional neural network* ……435
H. Kinjo, H. Ochi, T. Yamamoto (University of the Ryukyus, Japan)

C5-3 *Intelligent maneuvering of a flight-type wall-climbing robot* ……439
H. Miyagi, A. Nishi (Miyazaki University, Japan)

C5-4 *Block fuzzy neural networks for controlling an industrial manipulator* ……440
J. Tang, K. Kuribayashi (Yamaguchi University, Japan)

C5-5 *A fuzzy model to control the temperature in cooling of metal molds with spray robot* ……444
T. Sakamoto, K. Murakami (Ube Industries Co. Ltd., Japan)
K. Kuribayashi (Yamaguchi University, Japan)

S-C6: Advanced genetic algorithms for optimal network design (Invited Session) (Room C)

C6-1 *Improved genetic algorithm for generalized transportation problem* ……448
M. Gen, K. Ida, Y. Z. Li, J. Choi (Ashikaga Institute of Technology, Japan)

C6-2 *Genetic algorithms approach on leaf-constrained spanning tree problem* ……452
G. Zhou, M. Gen (Ashikaga Institute of Technology, Japan)

C6-3 *A genetic algorithm for bicriteria fixed charge transportation problem* ……456
Y. Z. Li, M. Gen, K. Ida (Ashikaga Institute of Technology, Japan)

C6-4 *A spanning tree-based genetic algorithm for reliable multiplexed network topology design* ……460
J. R. Kim, M. Gen, K. Ida (Ashikaga Institute of Technology, Japan)

S-D4: Artificial intelligence (General Session) (Room D)

D4-1 *Toward emergent intelligence in multiagent learning* ……464
K. Takadama, K. Shimohara (ATR, Japan)
T. Terano, (University of Tsukuba, Japan)
K. Hori, S. Nakasuka (The University of Tokyo, Japan)

D4-2 *Behaviors of pedestrians with learning ability in various underground areas* ……468
M. H. Zheng, Y. Kashimori, T. Kambara (The University of Electro-Communications, Japan)

S-D5: Genetic algorithms II (General Session) (Room D)

D5-1 *A framework for extending classifier systems under dynamic learning environments* ……472
R. Jiang, Y. Luo, D. Hu, H. Xi (Tsinghua University, China)

D5-2 *Quantum tunneling evolution: A model of life as a global optimization process* ……476
M. Hirafuji, S. Hagan (National Agriculture Research Center, Japan)

D5-3 *Automatic parallelization of sequential programs using genetic programming* ……480
C. Ryan, L. Ivan (University of Limerick, Ireland)

S-D6: Genetic algorithms III (General Session) (Room D)

D6-1 *A global search method for all roots of algebraic equations by genetic algorithm* ……484
S. Yamada, I. Yoshihara, K. Ozawa, K. Abe (Tohoku University, Japan)

D6-2 *Optimal control method using genetic algorithm and its application* ……488
T. Nishimura, K. Sugawara, I. Yoshihara, K. Abe (Tohoku University, Japan)

D6-3 *Optimization of foraging behavior by interacting multi-robots* ……492
K. Sugawara, I. Yoshihara, K. Abe (Tohoku University, Japan)

D6-4 *Optimization of delivery route in a city area using genetic algorithm* ……496
A. Takeda, S. Yamada, K. Sugawara, I. Yoshihara, K. Abe

B. (Tohoku University, Japan)

D6-5 *Time series prediction modeling by genetic programming without inheritance of model parameters* ……500
M. Numata, K. Sugawara, S. Yamada, I. Yoshihara, K. Abe (Tohoku University, Japan)

SE-4: Robotics intelligence and control I (Invited Session) (Room E)

E4-1 *Information transformation by virus-evolutionary genetic programming* ……504
N. Kubota, F. Kojima, S. Hashimoto (Osaka Institute Technology, Japan)
T. Fukuda (Nagoya University, Japan)

E4-2 *A description of dynamic behavior of sensory/motor systems with fuzzy symbolic dynamic systems* ……508
I. Takeuchi, T. Furuhashi (Nagoya University, Japan)

E4-3 *Intelligent fault tolerant system of vibration control for flexible structures* ……512
M. Isogai (Yamazaki Mazak Co., Japan)
F. Arai, T. Fukuda (Nagoya University, Japan)

E4-4 *An evolutionary technique for constrained optimization problems* ……516
M. M. A. Hashem, K. Watanabe, K. Izumi (Saga University, Japan)

E4-5 *Evolving in dynamic environments through adaptive chaotic mutation* ……520
D. P. T. Nanayakkara, K. Watanabe, K. Izumi (Saga University, Japan)

S-E5: Evolutionary computations (General Session) (Room E)

E5-1 *Organizational evolution by learning adaptive functions* ……524
M. Ishinishi, A. Namatame (National Defense Academy, Japan)

E5-2 *An Evolutionary design of commitment networks* ……528
K. Uno, A. Namatame (National Defense Academy, Japan)

E5-3 *The evolutionary approach for designing a virtual organization* ……532
Y. Shimoyama, A. Namatame (National Defense Academy, Japan)

E5-4 *What facilitates emergence of symbiosis* ……536
M. Chang, K. Ohkura, K. Ueda (Kobe University, Japan)

E5-5 *Synthetic collective behavior by multiple reinforcement learning agents in simulated dodgeball game* ……540
N. Ono, S. Yoshida (University of Tokushima, Japan)

E5-6 *Evolutionary design of analog electronic circuits* ……544
H. Shibata, S. Samadi, H. Iwakura (The University of Electro-Communications, Japan)

S-E6: Chaos (General Session) (Room E)

E6-1 *Periodic motion generated after chaos in a model of trading agents* ……548
M. Tanaka-Yamawaki, M. Tabuse (Miyazaki University, Japan)

E6-2 *A study of chaos associative memory* ……552
M. Nakagawa (Nagaoka University of Technology, Japan)

E6-3 *A chaotic synthesis model of vowels*
H. Koga, M. Nakagawa (Nagaoka University of Technology, Japan) ……556

E6-4 *A chaos model to solve the optimal stable marriage problem* ……560
R. Hiroi, M. Nakagawa (Nagaoka University of Technology, Japan)

S-A3: Perception in vision and hearing (Invited Session) (Room A)

A3-1 *Computation of motion direction by output neurons of the retina* ……564
H. Uchiyama (Kagoshima University, Japan)

A3-2 *Hysteresis phenomena in depth perception of a moving object* ……568
H. Jinnai, T. Kitazoe, T. Shii (Miyazaki University, Japan)

A3-3 *Visual perception depends on auditory stimuli?* ……572
K. Manabe, H. Riquimaroux (Doshisha University, Japan)

A3-4 *Speech recognition using stereovision neural network model* ……576
T. Kitazoe, S. I. Kim, T. Ichiki (Miyazaki University, Japan)

A3-5 *Pitch perception in the Japanese macaque: Physiological and behavioral approach* ……580
H. Riquimaroux, K. Manabe (Doshisha University, Japan)

A3-6 *Hands-free speech recognition in echo and noise environments* ……584
S. I. Kim. T. Kitazoe (Miyazaki University, Japan)

S-A4: Artificial brains etc. (General Session) (Room A)

A4-1 *Artificial mind in a classical context* ……588
S. Hagan, M. Hirafuji (N. A. R. C., Japan)

A4-2 *A computational model of viewpoint-forming process in searching solutions - theory and methods using hierarchical classifier system-* ……592
T. Yoshimi, T. Taura (The University of Tokyo, Japan)

A4-3 *An evolutionary architecture for a humanoid robot* ……598
P. Nordin, M. G. Nordahl (Chalmers University of Technology, Sweden)

A4-4 *A development of computer aided identification for systems under control* ……602
K. Oura, T. Murakoshi, K. Akizuki (Waseda University, Japan)
I. Hanazaki (Tokyo Denki University, Japan)

A4-5 *ATR's artificial brain (CAM-Brain) projict: A sample of what individual CoDi-1Bit model evolved neural net modules can do* ……606
H. D. Garis, N. E. Nawa (ATR, Japan)
M. Korkin (Genobyte Inc., USA)
F. Gers (Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Switzerland)
M. Hough (Stanford University, USA)

A4-6 *Spiker: Analog waveform to digital spiketrain conversion in ATR's artificial brain (CAM-Brain) project* ……610
M. Hough (Stanford University, USA)
H. D. Garis, N. E. Nawa (ATR, Japan)
M. Korkin (Genobyte Inc., USA)
F. Gers (Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Switzerland)

S-B7: Robotics intelligence and control II (Invited Session) (Room B)

B7-1 *Generation of jumping motion pattern for hopping robot using genetic algorithm* ……614
Y. Yoshida, T. Kamano, T. Yasuno, T. Suzuki (The University of Tokushima, Japan)
Y. Kataoka (Kataoka Machine Co., Ltd., Japan)

B7-2 *An evolutionary optimal obstacle avoidance method for mobile robots* ……618
M. M. A. Hashem, K. Watanabe, K. Izumi (Saga University, Japan)

B7-3 *Design and experiment of an omnidirectional mobile robot using fuzzy servo control* ……622
J. Tang, K. Sanefuji (Yamaguchi University, Japan)
K. Watanabe (Saga University, Japan)

B7-4 *An experiment on force control using fuzzy environment models* ……626
F. Nagata (Fukuoka Industrial Technology Center, Japan)
K. Watanabe, K. Izumi, K. Sato, S. Akama (Saga University, Japan)

S-B8: Artificial life (General Session) (Room B)

B8-1 *Realization of artificial human decision making based on conditional probability* ……630
M. Nakamura, S. Goto, T. Sugi (Saga University, Japan)

B8-2 *Artificial life with play instinct* ……636
S. Tamura, (Osaka University, Japan)

S. Inabayashi, Y. Kato (System Sogo Kaihatsu Co. Ltd., Japan)

B8-3 *Autocatalysis as internal measurement and origin of programs* ……640
S. Toyoda (Kobe University, Japan)

B8-4 *An immuno system model for generation of self tolerance and memory* ……644
Y. Ochi, Y. Kashimori, T. Kambara (The University of Electro-Communications, Japan)

B8-5 *Simulations of group action of artificial honey bees* ……648
Y. Niino (Johokagaku High School, Japan)
M. Sugisaka (Oita University, Japan)

S-C7: Soccer robotics (Invited Session) (Room C)

C7-1 *Robust and fast color-detecting using a look-up table* ……650
D. Y. Kim, H. K. Park, M. J. Chung (KAIST, Korea)

C7-2 *The multi-agent system's design based on behavior-based learning model* ……654
X. Wang, M. Sugisaka (Oita University, Japan)

C7-3 *An intelligent control strategy for robot soccer* ……658
T. Y. Kuc, S. M. Baek, I. J. Lee, K. O. Sohn (Sung Kyun Kwan University, Korea)

C7-4 *The design and development of robot soccer player* ……662
C. Zhang, C. Qu, D. Z. Gao (Northeastern University, China)

C7-5 *Simulation model of micro-robot soccer system* ……666
F. Rui, X. Liang, X. Xu (Northeastern University, China)

C7-6 *Role level design in a hybrid control structure for a vision-based soccer robot system* ……670
H. S. Shim, M. J. Jung, H. S. Kim, J. H. Kim, P. Vadakkepat (KAIST, Korea)

S-C8: Bio-Informatic Systems (Invited Session) (Room C)

C8-1 *Bio-Informatic coordination and interaction among artifacts and humans* ……674
O. Katai, T. Sawaragi (Kyoto University, Japan)

C8-2 *A basic study of virtual collaborator – the first prototype system integration* ……682
H. Ishii, W. Wu, D. Li, H. Ando, H. Shimoda, H. Yoshikawa
(Kyoto University, Japan)
T. Nakagawa (Mitsubishi Electric Corp., Japan)

C8-3 *Generating novel memories by integration of chaotic neural network modules* ……686
A. Sano (Kyoto University, Japan)

S-D7: Neural networks I (General Session) (Room D)

D7-1 *Constrained hierarchical path planning of a robot by employing neural nets* ……690
S. Patnaik, A. Konar, A. K. Mandal (Jadavpur University, India)

D7-2 *A new learning method using prior information of neural networks* ……694
B. Lu, K. Hirasawa, J. Murata, J. Hu (Kyushu University, Japan)

D7-3 *Design of neural network controller using feedback structure* ……699
W. J. Shin, S. Y. Lee (Kyungnam University, Korea)
K. Hirasawa (Kyusyu University, Japan)

D7-4 *Electrical equivalent circuit and resting membrane potential of neuron* ……703
X. Zhang, H. Wakamatsu (Tokyo Medical and Dental University, Japan)

D7-5 *An Adaptive associative memory system based on autonomous reaction between image memories* ……707
Y. Kinouchi, M. Mizutani, A. Satou, F. Shouji (Tokyo University of Information Science, Japan)
S. Inabayashi (System Sogo Kaihatsu Co., Ltd., Japan)

D7-6 *Human-face recognition using neural network with mosaic pattern* ……711
H. Kondo, S. B. A. Rahman (Kyushu Institute of Technology, Japan)

S-D8: Fuzzy control (General Session) (Room D)

D8-1 *Diagnosis with fuzzy belief networks* ……715
I. Chakraborty (Mie University, Japan)
A. Konar, A. K. Mandal (Jadavpur University, India)

D8-2 *Design of output feedback controllers for Takagi-Sugeno fuzzy descriptor systems* ……719
J. Yoneyama, K. Itoh, A. Ichikawa (Shizuoka University, Japan)

D8-3 *Stability study of fuzzy logic control system for an inverted pendulum* ……723
K. Nakano (Fukuoka Institute of Technology, Japan)
M. Tomizuka (University of California at Berkeley, USA)

D8-4 *Application of neural and fuzzy control strategies for a mobile vehicle* ……729
X. Wang, M. Sugisaka (Oita University, Japan)

S-E7: Neural networks II (General Session) (Room E)

E7-1 *The time minimum optimization strategy of idiotypic immune type signal transmission network system* ……733
H. Hirayama (Asahikawa Medical College, Japan)
Y. Okita (National Institute of Special Education, Japan)

E7-2 *An application of multi-modal neural network to multiple control system* ……737
T. Nakagawa, K. Sugawara, I. Yoshihara, K. Abe (Tohoku University, Japan)
M. Inaba (Hitachi Ltd., Japan)

E7-3 *Dynamical recognition via hybrid neural networks* ……741
N. Honma, K. Abe (Tohoku University, Japan)
H. Takeda (Tohoku Gakuin University, Japan)

E7-4 *Incremental evolution of neural controllers for navigation in a 6-legged robot* ……745
D. Filliat, J. Kodjabachian, J-A. Meyer (AnimatLab, France)

E7-5 *Pulse neural network applied to the binding problem-Implementation of decision making of the mobile robot-* ……751

M. Kojima, A. Yamaguchi, M.Kubo S. Mikami, M. Wada
(Hokkaido University, Japan)

E7-6 *Research on using dynamic neural networks in model predictive control* ……755
S. R. Li, F. Li (University of Petroleum, China)
Q. Lu (Tsinghua University, China)

S-E8: Micro-robot world cup soccer tournament (General Session) (Room E)

E8-1 *Adaptive positioning of soccer agents with hybrid learning system* ……759
N. Akiyama, K. Suzuki, M. Yamamoto, A. Ohuchi (Hokkaido University, Japan)

E8-2 *Recent design of motion planner and behaviors for soccer-playing robots* ……763
S. G. Hong, T. D. Eom, C. Y. Lee, M. S. Kim, J. J. Lee (KAIST, Korea)

E8-3 *Neural network architecture optimization and application using genetic algorithm* ……767
Z. J. Liu, M. Sugisaka (Oita University, Japan)

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Computer Simulation of Gene Flow in the Malaria Vector, Anopheles Gambiae, in West Africa

Charles Taylor

Abstract

One fundamental insight from Artificial Life has been to model populations of animals by populations of co-executing computer processes. We have been using such an approach to simulate the behavior of malaria-transmitting mosquitoes in West Africa. Mark-Release-Recapture experiments were performed, whereby several hundreds or thousands of mosquitoes were captured by hand, marked with fluorescent dust then released from a central location. On subsequent days new captures were made, and the location of marked mosquitoes was noted. At the same time these experiments were also simulated on the computer using artificial life systems -- Starlogo and Swarm -- and compared to the original data. We performed several rounds of simulation and experiment, adjusting the model and modifying the experiments to test assumptions. The goal has been to evaluate and guide future releases of genetically engineered mosquitoes in an effort to achieve better control of malaria in this region.

More detailed accounts of this work are given in

[1].1997. Carnahan, J. S-G. Li. C. Costantini. Y.T.Toure, and C. Taylor. *Computer Simulation of dispersal by Anopheles gambiae s.l. in West Africa*. pp. 387-394 in Artificial Life V. C.G. Langton and S. Shimohara, eds. MIT Press..

[2].1998. Tour, Y.Y, G. Dolo, V. Petrarca, S.F. Troar, B. Bouar, N. Sogoba, J. Carnahan, and C.Taylor. *Mark-Release-Recapture experiments with Anopheles gambiae s.l. in Banambanivillage*, Mali. Journal of Medical and Veterinary Entomology 12:74-83.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Discussion of "Living" Machines using Chaotic Control

Mark W. Tilden
Physics Division, Los Alamos National Laboratory
Email: mwtilden@lanl.gov

Abstract

Following several years of study into experimental Nervous Net (Nv) control devices, various successes and amusing failures have implied some general principles on the nature of capable control systems for autonomous machines and perhaps, we conjecture, even biological organisms. These systems are minimal, elegant, and, depending upon their implementation in a "creature" structure, astonishingly robust. Their only problem seems to be that as they are collections of non-linear asynchronous elements, only complex analysis can adequately extract and explain the emergent competency of their operation. Difficult, but the benefits are this could imply a cheap, self-programming engineering technology for autonomous machines capable of performing unattended work for years at a time, on earth and in space. Discussion, background and examples are given.

Introduction to Biomorphic Design

A Biomorphic robot (from the Greek for "of a living form") is a self-contained mechanical device fashioned on the assumption that chaotic reaction, not predictive forward modeling, is appropriate and sufficient for sustained "survival" in unspecified and unstructured environments. On the further assumption that minimal, elegant survival devices can be "evolved" from lesser to greater capabilities using silicon instead of carbon (using the roboticist as the evolutionary force of change). Over two hundred different "biomech" robots have been built and studied using solar power, motors, and minimal Nervous-Net control technology.

Nervous Networks (Nv) are a non-linear analog control technology that has been "evolved" to automatically solve real time control problems normally difficult to handle with conventional digital methods. Using Nv nets many sinuous robot mechanisms have been demonstrated that can negotiate terrain of inordinate difficulty for wheeled or tracked machines, as well as exhibiting very competent strategies for resolving immediate survival conundrums. The scale of devices developed so far has ranged from single "neuron" rovers to sixty neuron-distributed controllers with broad terrain abilities, and from machines under one-inch long to several meters in length. They have recognizable behaviors that, if not efficient, are at least sufficient to resolve otherwise intractable sensory integration problems. They remember, and more, use that knowledge to apply new strategies to acquire goals ("Living Machines", 1995).

This work has concentrated on the development of Nv based robot mechanisms by electronic approximations of biologic autonomic and somatic systems. It has been demonstrated that these systems, when fed back onto themselves rather than through computer-based control generators, can realistically mimic many of the abilities normally attributed to lower survival-biased biological organisms. That minimal non-linear systems can provide this degree of control is not so surprising as the part counts for successful Nv designs. A fully adept insect-walker, for example, can be fully controlled and operated with as little as twelve standard transistor elements.

The initial focus of Nv technology was to derive the simplest control systems possible for robotic "cradle" devices. The reason for this is threefold. First, such systems would feature robustness characteristics allowing inexpensive machines reliable enough to be trusted with performing unsupervised work in unstructured environments. Second, using Nv technology we hoped to resolve one of the most enviable things about biological designs, namely how nature can stick large numbers of lightweight, efficient actuators and sensors almost anywhere and still have them operate effectively. Third, and most important, exploration of minimal control systems may explain the biological paradox of why biological mechanisms can get by on so few active control elements. A common garden ant has roughly twenty-thousand control amplifiers distributed throughout its entire body, whereas a

digital watch may have as many as half a million amplifiers and still be unable to even walk. How does nature do so much with so little? The question is, what are the fundamental properties of living control systems, and what relationship do they have to the implicit abilities of Nv control topologies? Does Nv technology use some approximation of natural living things, is it the other way around, or is it neither?

Applications are now focusing on the use of this technology for adaptive survivor-based space hardware, and for use in unexploded ordinance, mines, and munitions detection and destruction. Interest and funding sources are JPL, DARPA, NASA, DOE, DOD, NIS and the Yuma Flats proving grounds.

Academic research is now concentrating on analysis of the non-linear characteristics of these systems, the development of an engineering lexicon, and several books on 'chaotic engineering', the science behind biomorphic robot construction.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Complexity and Control

Hidenori Kimura
The University of Tokyo
kimura@crux.t.u-tokyo.ac.jp

Keywords : uncertainty, control, model, capacity, learning, complexity

Control plays a role of increasing importance in various fields of modern technology to guarantee smooth and desired operations of real systems against the adverse effects of their environments. As the size and complexity of systems to be controlled increase, the task of control is getting more difficult and complex. Control system naturally becomes complex if the system to be controlled is complex and large-scale. Since control is always a task of on-line real time, the complexity of control is an important issue in design and implementation of real control systems.

Another cause of complexity of control systems is *uncertainty*. One of the major difficulties of the control of large complex systems lies in our lack of sufficient information and knowledge about the system to be controlled. Thus uncertainty is the issue of paramount importance in designing a satisfacroty control system. Guaranteering robustness against uncertainty tends to increase the complexity of controllers. In this paper, we discuss the complexity of controllers from the viewpoint of robustness. We present some of the recent results on the complexity of model set in relation to the complexity of robust controllers.

The algorithm of computing control signal from available sensing data is called *control law*. Control law is regarded as a mapping from available information to control signal. If the mapping is linear and time-invariant, the control law is called *linear time-invariant* (LTI), which is the simplest class of controllers. LTI controllers have been used extensively due to its simplicity in implementation and analytical tractability. Its limited ability, however, has motivated the use of *nonlinear time-varying* (NTV) controllers (e.g. adaptive control). Since NTV controllers are more complex than LTI controllers, it is interesting to know the advantages of NTV controllers over LTI controllers.

In 1984, a quantitative limitation of uncertainty was obtained, within which an LTI controller can stabilize uncertain systems. A surprizing result followed that this limitation is also valid for NTV controllers. In other words, it was shown that *NTV controllers have no advantage over LTI* controllers, as far as robust stabilizability of model set is concerned. Then, the problem arises: Suppose that the uncertainty is beyond the above limitation. Since NTV controllers don't work, what can we do to stabilize the plant ? One way is to reduce the uncertainty by collecting more information about the plant. Another way, and perhaps easier way, is to devide the original model set into pieces, each of which is smaller than the original one and is stabilizable by an LTI controller. Preparing a stabilizing LTI controller for each model set, we can stabilize the original model set by *switching* control law from one controller to another. This is a new type of adaptive control based on switching. The strategy of switching stabilization is well established by recent progress of hybrid control theory. As control process developes, we gradually identify a piece of the model set the real plant belongs. Thus, we can formulate control process as a process of *learning*.

The number of LTI controllers we have to prepare for stabilization of model set represents the complexity of the model set , as well as that of controller.
Actually, the problem of determining the smallest number of pieces that compose the model set is quite similar to the problem of finding the number of *coverings* of a class of functions. This is closely related to the notion of *capacity* introduced in statistical learning theory to represent the complexity of a class of functions. We present some results and open problems on this topics for further research.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Origins of Life: From Recent Topics

Tairo Oshima
The Department of Molecular Biology,
Tokyo University of Pharmacy and Life Science,
1432 Horinouchi, Hachioji, Tokyo 192-0392

Origins Life

Origins of life is still unsolved and extremely interesting subjects in Life Sciences. When and how did life begin on the primordial Earth? Nature succeeded, at least once, to assemble life in the primitive ocean from row materials such as amino acids, sugars and bases. Thus these questions are directly related to the design and creation of an artificial life. The author wishes to discuss some of recently developed investigations in this field.

Hot Origins of Life

Discovery of Archaebacteria (Archaea) and construction of a rooted phylogenetic tree[1-3] are the most significant progress in the recent years in the field of Biochemical Evolution. The rooted molecular tree indicted that the organisms belonging to the deepest branches are hyperthermophiles without any exception on both Archaeal and bacterial lineage[4]. This suggests that the primitive cells on the primordial Earth were also hyperthermophiles and life on our planet started in very hot oceans[5-6]. The surface temperature of the primordial Earth would be high due to green house effect of a large amount of carbon dioxide gas in the primitive atmosphere.

Ribozymes

After the discovery of ribozymes, a hypothesis of RNA origin of life became very popular, and the hypothetical world of living organisms consisted of only RNA molecules as essential cell components is called "RNA World". However, the hypothesis had some weak points; for instance, RNA is not stable at higher temperatures and ribozymes which are able to catalyze biopolymer synthesis has not been known. Recently Watanabe and his colleagues found that peptide bond synthesis can be catalayzed by ribosomal RNAs[7].

Genome Projects of Thermophiles

Thermopiles have generally smaller genomes[8]. So far whole genomes of a few hyperthermophilic Archaea and Bacteria have been sequenced. Though the functions of about half of structural genes are not clear yet, it will be possible to identify essential genes for life in the near future. It will be an interesting and important project to create a living cell with the smallest genome. Though out these studies, we will be able to define "life" in terms of genes and their functions. Genes required to live freely on our planet can be determined and the list of the essential genes will give us a new concept on life.

References

1. Woese, C.R., Kandler, O., and Wheelis, M. L. (1990) Towards a natural system of organisms: Proposal for the Domains Archaea, Bacteria, and Eukarya. Proc. Natl. Acad. Sci. U.S.A. 87, 4576-4579

2. Gogarten, J. P., Kibak, H., Dittrich, P., Taiz, L., Bowman, E. J., Bowman, B. L., Manolson, M. F., Poole, R. J., Date, T., Oshima, T., Konishi, J., Denda, K., and Yoshida, M. (1989) Evolution of Vacuolar

H+-ATPase: Implication for the Origin of Eukaryotes. Proc. Natl. Acad. Sci. U. S. A., 86, 6661-1115

3. Iwabe, N., Kuma, K., Hasegawa, M., Osama, S., and Miyata, T. (1989) Evolutionary Relationship of Archaeabacteria, Eubacteria, and Eukaryotes inferred from Phylogenetic Trees of Duplicated Genes. Proc. Natl. Acad. Sci. U. S. A., 86, 9355-9359

4. Stetter, K. O. (1993) Life at the upper temperature border. In "Frontier of life" ed. By Tran Thanh Van, J., Tran Thanh Van, K., Mounolon, J. C., Schneider, J., and McKay, C., pp.195-219, Editions Frontieres, Gifsur-Yvette

5. Yamagishi, A. and Oshima, T. (1995) Return to Dichotomy: Bacteria and Archaea. in "Chemical Evolution: Self-Organization of the Macromolecules of Life" ed. By Chela-Flores, J., Chadha, M., Negron-Mendoza, and Oshima, T., pp. 155-158, A. Deepak Publishing, Hampton, Virginia, USA

6. Pace, N. (1991) Origin of Life-Facing Up to the Physical Setting. Cell, 65, 531-533,

7. Nitta, I., Kamada, Y., Noda, H., Ueda, T., and Watanabe, K. (1998) Reconstitution of Peptide Bond Formation with Escherichia coli 23S Ribosomal RNA domains. Science, 281, 666-667

8. Yamagishi, A. and Oshima, T. (1998) Sulfolobus genome. In "Bacterial Genomes: Physical Structure and Analysis", ed. deBrujin, F., Lupski, J. R., and Weinstock, G. M., Chapman & Hall, New York

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

The mind of the brain

Gen Matsumoto

The Institute of Physical and Chemical Resarch

e-mail: gen@brainway.riken.go.jp

Abstract: The brain works as a system which is capable not only of acquiring information processing algorithm automatically, but also of selecting the informations to be processed by itself. To elucidate strategic algorithms for the brain to acquire those algorithms leads to certify how the brain grows by itself. This also means to elucidate what factors are essential for our mental growth, and therefore, at least partly, to answer to a question of "what human is". Further, it enables to develop a novel information processing system as an engineering realization of Brainway computer; its operation principle is the same as the brain's. That is to say, the Brainway computer can select information to process by itself, and also acquire the algorithm for the processing of the selected information by itself. Realization of Brainway computer turns to conform the brain operation principle and as well, Brainway computer is considered ultimately supplementary to an existing type of computer that works following to its programs. In this paper, we introduce brain science of which stresses its stand point on "Creating the Brain", and at the same time, about Brainway computer inclusive of its research and development as engineering counterpart of the brain.

1. Introduction - Two streams of research on the brain and recent advancement of research and development on computers –

The brain is a system which selects information by itself and acquires also by itself the algorithm, how to processinformation. The brain scientific research in general mainly aims at elucidation of the algorithm which the brain has acquires. Since the brain's acquired algorithm is viewed as being expressed in its three dimensional structure of a neuronal circuit as well as in its activity, the brain science follows an established method and principle of materialistic science to analyze the structure and activity which is a so called approach "Understanding the Brain". As an concrete example of this brain science for "Understanding the brain", it tries to answer to a question why Japanese people can speak Japanese language. When we find out some reasoning for this, we come to know the algorithm that enables us to speak Japanese. Accordingly, we could apply the algorithm as this scientific product to development a software for a conventional program-operated computer, resulted in realizing the computer that can speak Japanese. As such for natural application of a stream of this brain research approach to computer engineering, it should be largely contributed to the development of software, a technique to utilize the existing computer (see the graph.1). However, this research is not lead us to realize Brainway computer, a novel concept of computer that is operated by means of the of brain principle for its algorithm self-acquisition. On the contrary, there is another approach of brain science which understands the brain by clarifying its strategic algorithm how the brain acquires its algorithms. This is exemplified in what Japanese person grows up to be a Japanese speaker due to his/her surrounded circumstance in which people speak in Japanese, though without knowing why he/she can speak Japanese. This is an approach of brain science what so called "Creating the Brain", and looks superficially to belong to a stream of the research on neural network hitherto. Here, it should be noted that the research on neural network sets up its main

research objective on realizing an automatic algorithm acquisition system in terms of engineering, and that brain science is merely giving one of the methods for this. On the contrary, in brain science of "Creating the Brain", the aim and method are set opposite way round, quite contrasted with the ones as the neural network research has long persued. That is, the engineering realization Brainway computer operated by the theory of the brain is a mean for us to carry out the objective of "Creating the Brain", and this would lead us to verify the brain principle how the brain can acquire its algorithms by itself and eventually to understand the brain's acquired algorithms. For the latter, we could know the brain's algorithms by studying the brainway computer's acquired algorithms in details which should be expressed in the computer more understandable for us than those in the brain. This would be quite easily understood when we think about how we came to find out the principle of flying over the sky through two different approaches; one to study on propulsion birds and another to develop an airplane. It should be noted that the principles for flying over the sky, namely lifting and propulsion powers, was elucidated mainly due to hydrodynamical physics which actually aimed at development of an airplane. The difference is their structural and functional expressions, caused by their different objectives to fly over the sky, though their principles of flying in bird and airplane are the same. The fundamental requirements for airplanes should be put stress on mass transportation, speed, safety and others, which are quite contrasted with the bird's requirement Further, different structural. materials are also required for each constructions. In aeronautics, the operating force was, as a consequence, expressed as propeller or jet-engine, completely different in idea from bird's fluttering. Similarly, the research for the engineering realization of Brainway computer is essential for us to understand the brain principle through our research efforts of "Creating the Brain" which should be another brain science to understand the brain. Trough this research, we will establish an engineering-realized image of a brainway computer, and it will eventually be realized as a revolutionary engineering product which has an ability to manage soft information processing quite contrasted with the existing computer of hard information processing. Here it is notable that what most importantly birds have contributed to aeronautics was the fact that birds have been actually flying over the sky. The idea of flying over the sky was brought to the people from this living proof, and it was really developed and realized by those who believed to be able to engineer something that could fly over the sky just like birds. Identically, it should be the most important factor in realization of brainway computer that we have our brain, as a living proof of the system for information processing that can acquire its algorithm by itself. As the research of "Understanding the Brain" is being unfolded further, it is expected to lead us towards an understanding of mind, in the sense of consciousness, will, reason and knowledge, etc., as phenomena exclusive to the activities of the brain.

In a word, this is an understanding of mind from the brain on the brain scientific base. On the contrary, the research of "Creating the Brain" aims at understanding of the mind as something existing behind the brain, but actually generating and operating the brain. In a word, we can say this is an elucidation of mind of the brain. We believe that these two approaches of brain science works together to bring us ingeniously scientific product and eventually lead to profound understanding of the brain as well as an advancement of computer engineering(see the graph.1). In other words, it is very crucial to line up in parallel and promote these two approaches side by side. We often hear an opinion saying that the research on the brainway computer should await to start its promotion until the research on understanding the brain is advanced further. Understanding of the brain here means to understand the brain in the sense of the approach "Understanding the Brain". It is however possible to propel the research on the brain-style computer without any knowledge gained from the research of "Knowing the Brain". This is because we acquire the ability to speak Japanese without knowing why, or, in terms of "Understanding the Brain", without understanding anything about the algorithms which our brain acquires by itself. Accordingly, the brainway computer should be able to acquire the algorithms by itself and speak Japanese auto-

matically without knowing the reason. On the other hand, elucidation of the algorithms the brainway computer has acquired would be far easier than that of the algorithms of the brain itself. In this sense, the research of "Creating the Brain" can partially covers the one of "Understanding the Brain". On the contrary, the latter contributes to show the goal of the brain's acquiring algorithms which the former should aim at. This is because the research of "Understanding the Brain" would equally represent the algorithms that are supposed to be conveyed by the research of "Creating the Brain". In a word, both of "Understanding the Brain" and "Creating the Brain" are two essentials for the understanding of the brain.

2. What kind of computer the brain is: the brain information principle

The brain information principle of the brain is represented as its memory-based architecture. Here, the word, memory, does not mean data storage as often refereed in terms of a processor based architecture computer with which stored in memory for the processing to be changeable according to the program. Memory here used is nothing but solidification of the brain's acquired algorithms. The brain algorithms are expressed as neuronal and glial structure and as well, their activities. When the structure is once solidifies, the activity adherent to the structure can be activated for non-destructive memory retrieval. Then, learning effect takes place only when the brain put outputs; in consequence, the neuronal circuit alters its structure and accordingly changes the activity. In this sense, the memory is dynamically changeable. The detailed characteristics of the brain's memory-based architecture computer are described below.

3. The brain's information principle

Here we should answer to a question why the brain is a memory-cased architecture.

(1) Solidification (memorization) of the algorithms comes first

First of all, the brain acquires to solidify its algorithms. The brain's strategic algorithm for acquiring algorithms by itself is called learning algorithm. The brain's acquired algorithms, once solidified as a long-term memory, can be non-erasable. Apparent threshold level for memory retrieving becomes higher, as time passes after its is solidified, but the memory itself remains. Due to this memory characteristics, the memory can be harder and harder for its retrieval with time.

(2) Input information is index for memory retrieval

Information input to the brain is utilized as a kind of index to search an algorithm (memory) already solidified in the brain for its output. That is, any output from the brain is resulted as selection of memory which has been stored before the input. This is quite contrasted with the conventional computer with processor-based architecture where the input information is sequentially processed to get an output.

(3) Output decision is made only when necessary

For the input information to index memory the brain roughly but nimbly grasp its semantic metric, to evaluate its value as to whether it is valuable for the brain to respond to it or not, accordingly. That is, the brain searches for memory to make output only when it recognizes its value, but no output is put when judged not worth doing so.

(4) Memory changes as the brain outputs

When the brain outputs, learning effect takes place to make a change to the algorithm of the brain; in other words, the memory is changeable. That is to say, the learning algorithm is output-dependent. Further, it is time-sequential-event associated (1). The latter characteristic puts the brain to have an ability to predict things or matters taking place in some future concerning those currently processed. The information processing system of the brain works in order of (1)→(2)→(3)→(4)→(1)→ - - -, as described above. The brain gets "recognition" about the matters it currently concerns with when its prediction copes with the algorithm modified in the separated process (1)→(2)→(3)→(4)→(1)- - - ; but if not, it experiences no feelings of understanding nor recognition. From this point of view, the information processing systems in the brain will be characterized in more details by comparison with the existing computer that is operated by program (in the rest, abbreviated just as computer) (see Figure 2).

The most fundamental difference between these is on their objectives as the information processing systems: the objective of computer is to get output; on the contrary, that of the brain is acquisition and formation of the algorithm, outputting being simply a mean for it. The objective (or, aim) of the computer is to get output by processing input information as ordered, in accordance with the purpose and processing procedure nimbly as programmed. On the other hand, the objective of the brain is the algorithm acquisition and its building-up; for the brain, outputting is just a mean to carry out its mission. This information processing characteristic of the brain is common among all biological information systems including genetic information (see Figure 2). In order to attain this aim, the principle of the brain is characteristically expressed as a memory-based architecture (see Graph 2):

(1) Automatic acquisition and formation of the brain's algorithm

In contrast, the computer sequentially processes input information to get output. For the sequential processing, it is essential to set up a program in the computer beforehand, which orders the computer how the input information should be processed and for what the program is purposed. On the contrary, the brain works without being externally programmed, and its outputting effects on formulating and renewing a program or an algorithm.

(2) Time for output is almost equivalent to memory-indexing time

When receives input information, the brain first acknowledges its semantic meaning, fast but roughly, and then, judges its value: this is to set up a condition in advance for memory indexing. Then it searches out the memory most adequate to input information, to output the algorithm. The condition is set up by limbic system in order to establish a domain for memory indexing in parieto-temporal association cortex. It restricts an area of cerebral neocortex to index the memory within. In parallel to designate the memory area for targeting the neocortex, more detailed analysis regarding the cognitive processing is going on in sensory association areas of cerebral neocortex. This analysis is like a kind of refined filtering. Finally it generates an external. Consequently, the time for refined filtering in the neocortex is much longer than that required for advance indexing. This is the basic structure for the brain knowledge-based information processing that includes such as pattern recognition. By virture of this memory-based architecture, it takes only of the order 100 ms to recognize a human face, is spite of the situation that several to 10msec is needed for neural information processing in a single neuron. For the existing computer of sequential processing, the required time for outputting is a product of device speed(memory and processor speed) and a number of processing steps. It therefore takes quite long time to process face recognition even if could be done by the existing supercomputer. In fact, the current computer processor based-architecture cannot recognize human face, as raised with reasons shown below in (3). Based on this characteristic of memory-based architecture, when the basic brainway device is developed, which is the fundamental device for the brainway computer, equivalent to neurons for the brain, then it would be realized as a computer which can recognize a human face, say, a million times as faster as the brain when a device speed is a million time faster than that of neuron.

This has been basically certified by the development of the brainway device materialized as silicon LSI. Further, it should be less time required to establish an algorithm for memorizing a face, that is, time required for learning, since it can generate to output within a much less time according to the theory of output-dependent algorithm for learning. In a word, the brain and brainway computer can output outstandingly faster than the existing computer. However, fast processing does not necessarily mean correctness. This is due to the fact that the objective of the brain and brain-way computer is lied on its acquisition and formation of algorithms, so that outputting is just a mean to carry out its objective. But it could be also said that it will be able to manage to process the algorithm more promptly and accurately by repeating outputting, and eventually present more precise answer to it.

(3) Soft information processing

The brain can correspond most adequately to input information by outputting when it decides necessary to do so.

This particular property leads to us to say that the brain processes information "softly". As the brain outputs, the algorithmic memory automatically changes according to the learning algorithm. That is to say, the brain grows up, or, the brain creates itself. It has been shown in our paper that the output-dependent learning algorithms hold for both local and global areas of the brain. On the contrary, conventional computers can process input information sequentially for output as programmed, but never has an ability to manage anything that has not been programmed. In this sense, current computers process information "hard". Computers merely reacts but not corresponds to input information. Computers could do anything. For this, the program is prerequisite for it. However, it would be almost impossible to prepare the program for all the matters in the real world. As a result, it can only be an limited address available at the greatest common measure. This is why artificial intelligence could not be well realized in its own term as originally aimed at. In this age of information society, both people and society are, whether or not they are conscious manualized in order to match them to this computer. Further, the hard computer is really inadequate to crisis or emergency management. This is also because, unlike the brain, it cannot predict the future based on the past experience. Let think about the case that we will try to develop a car which can safely drive by itself. Even with the conventional computer, we could develop a car that can drive automatically and safely on a test driving road. We could prepare program for the car since happenings on the test driving road could be well predicted. However, it is not the case in the centre if a down town, where some accidental happenings might easily take place. The program could be mostly made, only after some accident really happens. Further, the programming has a tendency to generalize the case into the greatest common measure. For this general tendency, when an information society is managed with the hard computer, it would force people to follow the program and result in giving too excess restriction on them: it rather pushes the society hard for people to live. This is actually represented in life regulated by social discipline such as low. Furthermore, if social security is served by the hard computer, although it may decrease a percentage of accidents in the extent to which the computer program or manual is well equipped to deal with them, it could cause to enlarge accidents in size. On the contrary, the brain can deal with cases the most appropriately in order to hold the damage at minimum. As one experience is conceptualized and synthesized in order to be able to apply for other cases, it could give a better solution when similar cases happen. This is then memorized as a generalized ability. There would be many more examples regarding how the hard computer brings inappropriateness both to the society and human beings, when applied to them as a tool for communication and information processing. In this age of information society being rapidly developed, it should be noted that necessity and emergency to develop the brainway computer would be well recognized.

(4) A large scale problem could be dealt with by hierarchically structured memory and add-on-learning

The brain memorizes an algorithm in a long-term fashion, and output is made based on this algorithm. This results in letting the brain acquire a renewed algorithm. That is to say, the brain has an ability of add-on-learning. This ability of add-on-learning by means of the hierarchical structuring enables the brain to deal with a large scale problems. That is, memory once acquired in the past in a long term fashion can never be erased then, but instead, restructured together with a memory newly acquired. As the newly acquired memory is put into the structure, the past memory seems to be forgotten since it becomes difficult to be retrieved as its threshold level becomes higher as time goes on. However, once memorized in a long term fashion, it is non-erasable. The old memory is recalled and output only in either of cases when more or less the same situation is recapitulated as the memory was originally input, or, when the brain activities are greatly enhanced for some reasons. In the former case, it outputs especially the memory related to matters happened in the past,

while in the latter, memories are non-selectively output at a time-sequential manner just like pictures of kaleidoscope. In contrast, the computer could only deal with a large scale problem with its large-scale program. Consequently, a large scale problem could be only handled with development of correspondingly a large scale software. This causes what so-called "software crisis".

(5) A variety of neuronal cells not needed as the brain's fundamental device

In the brain, there are four kinds of neuronal cells in the cerebellum, and about ten kinds in the cerebrum, respectively. As these are connected peculiarly and form a specific structure of neuronal circuits, the brain can process information. When these neuronal cells are realized as brainway device of the brain's fundamentals, it gains an ability to choose and make a decision as to how to process information, and as well, learn and memorize which is supposed to be done basically by a single device. The fundamental characteristics such as calculation of the fundamental device are automatically acquired by learning, and it is also capable of an ability that can process things in parallel, for instance an ability with which one neuronal cell can calculate 10000 of input connections at a real time, etc. It is contrastive when considered the fact that, in order to shorten the time for calculation of the existing computer, it is necessary not simply to structure it to fasten the speed of processor and memory circle but also to advance revolutionarily the ability of processor for calculation. The brainway device's ability for calculation is acquired by the device from learning, so that a designer just has to concentrate at most on setting up the ability. Moreover, in the world of the existing processor-based architecture, processor and memory are structured to be placed separately in space, so that they have been materialized as different chips respectively. In contrast, as the brainway device is structured in a memory-based architecture, it is favourable that the processor and memory are placed close to each other on the same siliconware. The research on this technology of the device has just been started.

(6) Brain's low-consumption of energy

The brain activates itself as much as it is necessary, for outputting of the memory and forming and renewing of the algorithm. This means it consumes exceedingly a small amount of energy. On the contrary, the computer, as typically seen in a supercomputer, is causing a serious problem with its increasing amount to consume energy and heat that accompanies. Larger the scale for information processing increases the balance of energy consuming between those two. This means that even if aupercomputer that has an ability to simulate the whole activities of the brain is realized in theory, it should be very doubtful in reality whenconsidered from this point of view. In conclusion, , the brainway computer is inevitable as a tool to understand the brain. To solve the problems and realize the brain's information processing as described above is the objective set up for the research of "Creating the brain", and resulted in producing a novel concept of information processing system "Brainway Computer".

Figure.1

Two streams of Research on Brain and recent Advancement of Research and Development of Computer

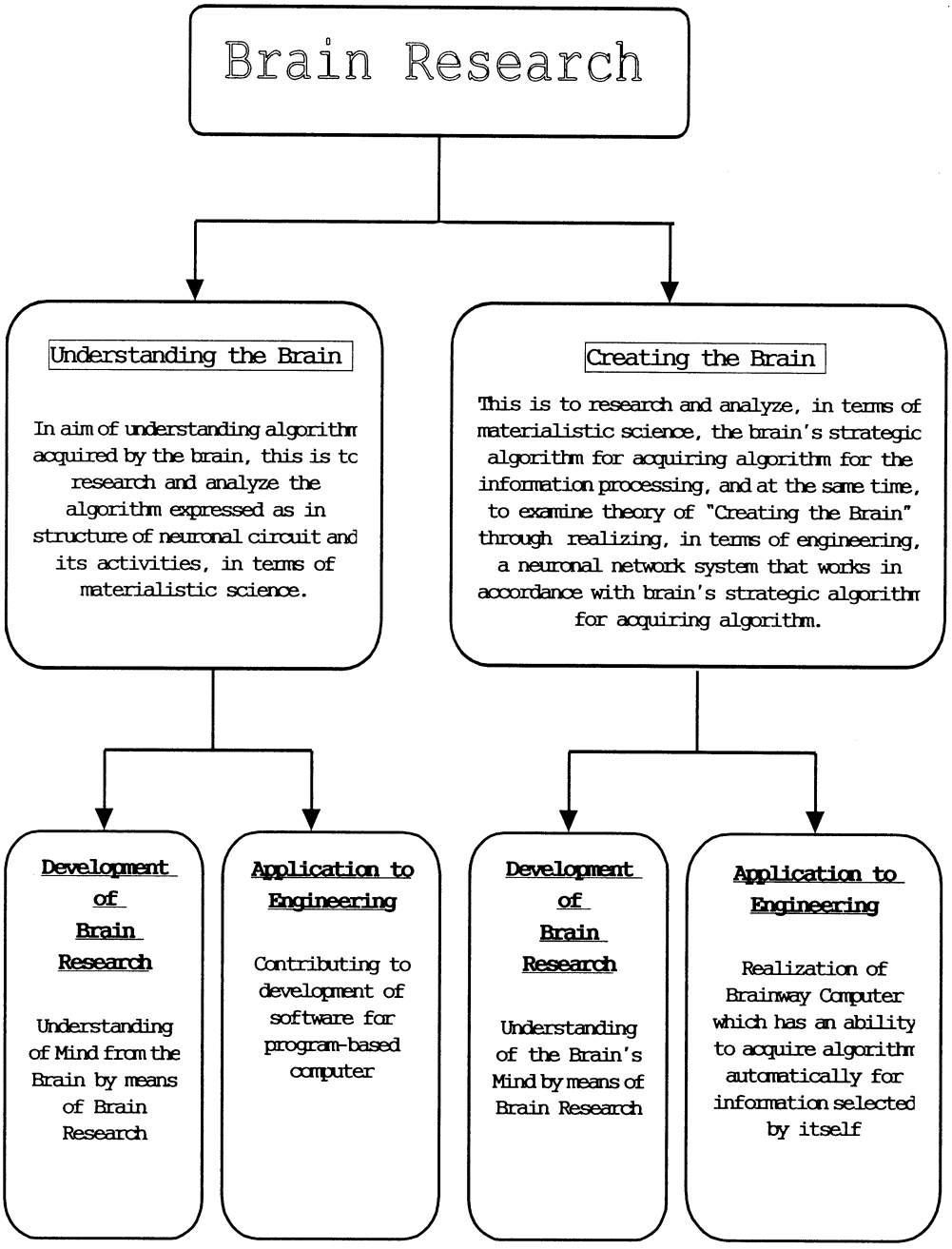
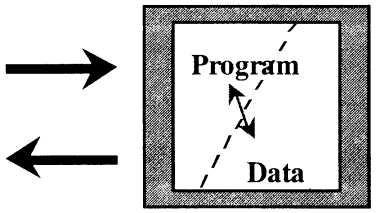
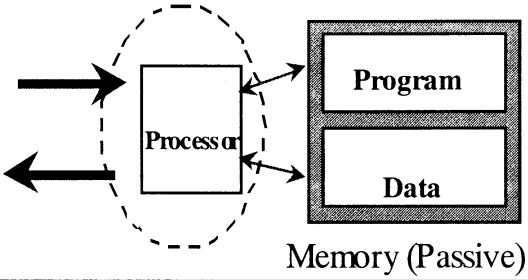


Figure 2:

Brain: Memory-based Architecture
VS
Von Neumann Computer

Brain: Memory-based Architecture	Processor-based Architecture (Von Neumann Computer)
 Memory (Active)	 Memory (Passive)
Automatic acquisition of program and data	Aim and procedure for information processing given as program
Time for Indexing Memory ≒Time for outputting	Sequential processing: Time for calculation = Speed of Device times numbers of steps
Soft Information Processing Able to make appropriate reaction against input information when necessary A Processing system which reacts against matters judged valuable by the brain	Hard Information-processing Reacts only against what being prepared in program Manuals to be fully equipped in order to insure the computer's mainframe for calculation for all the matters to be processed Unsuitable for management of crisis/emergency
Large-scale Accident Dealt with by Hierarchically structured memory and add-on-learning ability	Large-scale Problem Increases demand for development of software (Software Crisis)
Hardware: basically dealt with a single basic device	Hardware: A Variety of Processors
Low-comsumption of Energy and low-production of Heat : only an amount required to activate a particular part of the brain	Increases amount of Energy-Comsumption and Low-production of Heat: Serious Problem accompanied with Speeding up Processor

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

BizSim: The World of Business in a Box

John L. Casti
Santa Fe Institute
1399 Hyde Park Road
Santa Fe, NM 87501, USA

Abstract

The world of business is one especially well-suited for study by agent-based modeling and simulation. The agents in such simulations can range from traders in a financial market to corporations in a large-scale industry. This paper looks at three examples of such simulations—a stock market, the world's catastrophe insurance industry and a supermarket—as illustrations of how such methods can be used to examine questions in the world of commerce.

1 The Computer as a Laboratory

The central process distinguishing science from its competitors—religion, music, literature, mysticism—in the reality-generation business is the so-called *scientific method.* An integral part of this method by which we arrive at scientific "truth," is the ability to do controlled, repeatable laboratory experiments by which hypotheses about the phenomenon under investigation can be tested. It is just such experiments that on a good day lead to the theories and paradigms constituting today's "scientific" world view. And, more than anything else, it is the inability to perform experiments of this type that separate the natural sciences from the worlds of social and behavioral phenomena. In the latter, we have no way of doing the experiments necessary to create a bona fide scientific theory of processes like stock market dynamics, road-traffic flow, and organizational restructuring.

In an earlier, less discerning era, it was often claimed that the realm of human social behavior was beyond the bounds of scientific analysis, simply because human beings are "complex", "unpredictable", "display free will", "act randomly", and so on and so forth. It's hard to believe that any modern system theorist would do anything but laugh at such childish and naïve attitudes to the creation of workable and worthwhile *scientific* theories of social and behavioral phenomena. The major barrier to bringing the social beneath the umbrella of science is not the non-explanations just given in quotes, but the fact that until now we have had no way to test hypotheses and, therefore, make use of the scientific method in the creation of theories of social behavior. Now we do. And that laboratory in which we do our experiments is the digital computer. Let me illustrate with an example from the world of finance.

2 Booms and Busts, Bubbles and Crashes

In the fall of 1987, W. Brian Arthur, an economist from Stanford, and John Holland, a computer scientist from the University of Michigan, were sharing a house in Santa Fe while both were visiting the Santa Fe Institute. During endless hours of evening conversations over numerous beers, Arthur and Holland hit upon the idea of creating an artificial stock market inside a computer, one that could be used to answer a number of questions that people in finance had wondered and worried about for decades. Among those questions were:

- Does the average price of a stock settle down to its so-called *fundamental value*—the value determined by the discounted stream of dividends that one can expect to receive by holding the stock indefinitely?
- Is it possible to concoct technical trading schemes that systematically turn a profit greater than a simple buy-and-hold strategy?
- Does the market eventually settle into a fixed pattern of buying and selling? In other words, does it reach "stationarity"?
- Alternately, does a rich "ecology" of trading rules and price movements emerge in the market?

Arthur and Holland knew that the conventional wisdom of finance argued that today's price of a stock is simply the discounted *expectation* of tomorrow's price plus the dividend, given the information available about the stock today. This theoretical price-setting

procedure is based on the assumption that there is an objective way to use today's information to form this expectation. But the information available typically consists of past prices, trading volumes, economic indicators, and the like. So there may be many perfectly defensible ways based on many different assumptions to statistically process this information in order to forecast tomorrow's price. For example, we could say that tomorrow's price will equal today's price. Or we might predict that the new price will be today's price divided by the dividend rate. And so on and so forth.

The simple observation that there is no single, best way to process information led Arthur and Holland to the not-very-surprising conclusion that deductive methods for forecasting prices are, at best, an academic fiction. As soon as you admit the possibility that not all traders in the market arrive at their forecasts in the same way, the deductive approach of classical finance theory, which relies upon following a *fixed* set of rules to determine tomorrow's price, begins to break down. So a trader must make assumptions about how other investors form expectations and how they behave. He or she must try to psyche out the market. But this leads to a world of *subjective* beliefs and to beliefs about those beliefs. In short, it leads to a world of induction in which we generalize rules from specific observations rather than one of deduction.

In order to address these kinds of questions, Arthur, Holland and their colleagues constructed an electronic stock market, in which they could manipulating trader's strategies, market parameters, and all the other things that cannot be done with real exchanges. The traders in this market are assumed to each summarize recent market activity by a collection of descriptors, which involve verbal characterization like "the price has gone up every day for the past week," or "the price is higher than the fundamental value," or "the trading volume is high." Let us label these descriptors A, B, C, and so on. In terms of the descriptors, the traders decide whether to buy or sell by rules of the form: "If the market fulfills conditions A, B, and C, then buy, but if conditions D, G, S, and K are fulfilled, then hold." Each trader has a collection of such rules, and acts in accordance with only one rule at any given time period. This rule is the one that the trader views as his or her currently most accurate rule.

As buying and selling goes on in the market, the traders can reevaluate their different rules by assigning higher probability of triggering a given rule that has proved profitable in the past, and/or by recombining successful rules to form new ones that can then be tested in the market. This latter process is carried out by use of what is called a genetic algorithm, which mimics the way nature combines the genetic pattern of males and females of a species to form a new genome that is a combination of those from the two parents.

A run of such a simulation involves initially assigning sets of predictors to the traders at random, and then beginning the simulation with a particular history of stock prices, interest rates, and dividends. The traders then randomly choose one of their rules and use it to start the buying-and-selling process. As a result of what happens on the first round of trading, the traders modify their estimate of the goodness of their collection of rules, generate new rules (possibly), and then choose the best rule for the next round of trading. And so the process goes, period after period, buying, selling, placing money in bonds, modifying and generating rules, estimating how good the rules are, and, in general, acting in the same way that traders act in real financial markets.

A typical frozen moment in this artificial market is displayed in Figure 1. Moving clockwise from the upper left, the first window shows the time history of the stock price and dividend, where the current price of the stock is the black line and the top of the grey region is the current fundamental value. The region where the black line is much greater than the height of the grey region represents a price bubble, whereas the market has crashed in the region where the black line sinks far below the grey. The upper right window is the current relative wealth of the various traders, and the lower right window displays their current level of stock holdings. The lower left window shows the trading volume, where grey is the number of shares offered for sale and black is the number of shares that traders have offered to buy. The total number of trades possible is then the smaller of these two quantities, because for every share purchased there must be one share available for sale.

After many time periods of trading and modification of the traders' decision rules, what emerges is a kind of ecology of predictors, with different traders employing different rules to make their decisions. Furthermore, it is observed that the stock price always settles down to a random fluctuation about its fundamental value. However, within these fluctuations a very rich behavior is seen: price bubbles and crashes, market moods, overreactions to price movements, and all the other things associated with speculative markets in the real world.

The agents in the stockmarket simulation are individual traders. A quite different type of business sim-

ulation emerges when we want to look at an entire industry, in which case the agents become the individual firms constituting that industry. The world's catastrophe insurance industry served as the focus for just such a simulation exercise called *Insurance World,* carried out by the author and colleagues at the Santa Fe Institute and Intelligize, Inc. over the past couple of years.

3 Insurance World

As a crude, first-cut, the insurance industry can be regarded as an interplay among three components: *firms,* which offer insurance, *clients,* who buy it, and *events,* which determine the outcomes of the "bets" that have been placed between the insurers and their clients. In Insurance World, the agents consist of primary casualty insurers and the reinsurers, the firms that insure the insurers, so to speak. The events are natural hazards, such as hurricanes and earthquakes, as well as various external factors like government regulators and the global capital markets.

Insurance World is a laboratory for studying questions of the following sort:

- *Optimal Uncertainty:* While insurers and reinsurers talk about getting a better handle on uncertainty so as to more accurately assess their risk and more profitably price their product, it's self-evident that perfect foreknowledge of natural hazards would spell the end of the insurance industry. On the other hand, complete ignorance of hazards is also pretty bad news, since it means there is no way to weight the bets the firms make and price their product. This simple observation suggests that there is some optimal level of uncertainty at which the insurance—but perhaps not their clients—can operate in the most profitable and efficient fashion. What is that level? Does it vary across firms? Does it vary between reinsurers, primary insurers, and/or end consumers?

- *Industry Structure:* In terms of the standard metaphors used to characterize organizations—a machine, a brain, an organism, a culture, a political system, a psychic prison—which type(s) most accurately represents the insurance industry? And how is this picture of the organization shaped by the specific "routines" used by the decisionmakers in the various components making up the organization?

The simulator calls for the management of each firm to set a variety of parameters having to do with their desired market share in certain regions for different types of hazards and level of risk they want to take on, as well as to provide a picture of the external economic climate (interest rates, likelihood of hurricanes/earthquakes, inflation rates and so forth). The simulation then runs for 10 years in steps of one quarter, at which time a variety of outputs can be examined. For instance, Figure 2 shows the market share for Gulf Coast hurricane insurance of the five primary insurers in this toy world, under the assumption that the initial market shares were *almost* identical—but not quite. In this experiment, firm 2 has a little larger initial market share than any of the other firms, a differential advantange that it then uses to squeeze out *all* the other firms at the end of the ten-year period. This is due to the "brand effect," in which buyers tend to purchase insurance from companies that they know about.

As a final example of what simulation and business have to say to each other, consider the movement of shoppers in a typical supermarket. This world is dubbed *SimStore* by Ugur Bilge of SimWorld, Ltd. and Mark Venables at J. Sainsbury in London, who collaborated with the author on its creation.

4 SimStore

The starting point for SimStore is a real supermarket in the Sainsbury chain, one located in the London region of South Ruislip. The agents are individual shoppers who frequent this store. These electronic shoppers are dropped into the store, and then make their way to the various locations in the store by rules such as "wherever you are now, go to the location of the nearest item on your shopping list," so as to gather all the items they want to purchase.

As an example of one of the types of outputs generated by SimStore, customer checkout data are used to calculate customer densities at each location. Color codes are with descending order: blue, red, purple, orange, pink, green, cyan, grey and nothing. Using the Manhattan metric pattern of movement, in which a customer can only move along the aisles of the store, all locations above 30 percent of customer densities have been linked to form a most popular customer path. Once this path is formed a genetic algorithm will minimize (or maximize!) the length of the overall shopping path.

In the same store, this time each individual customer path has been internally calculated using the simple "nearest neighbor" rule noted above. All customer paths have been summed for each aisle, in order

to calculate the customer path densities. These densities are displayed in Figure 3 as a relative density map using the same color code just mentioned.

5 Simulation is Good for Business

Large-scale, agent-based simulations of the type discussed here are in their infancy. But even the preliminary exercises outlined here show the promise of using modern computing technology to provide the basis for doing experiments that have never been possible before. Even better, these experiments are exactly the sort called for by the scientific method—controlled and repeatable—so that for the first time in history we have the opportunity to actually create a *science* of human affairs. If I were placing bets on the matter, I'd guess that the world of business and commerce will lead the charge into this new science that will form during the 21st century.

References

[1] Casti, J., *Would-Be Worlds*, John Wiley & Sons, 1997.

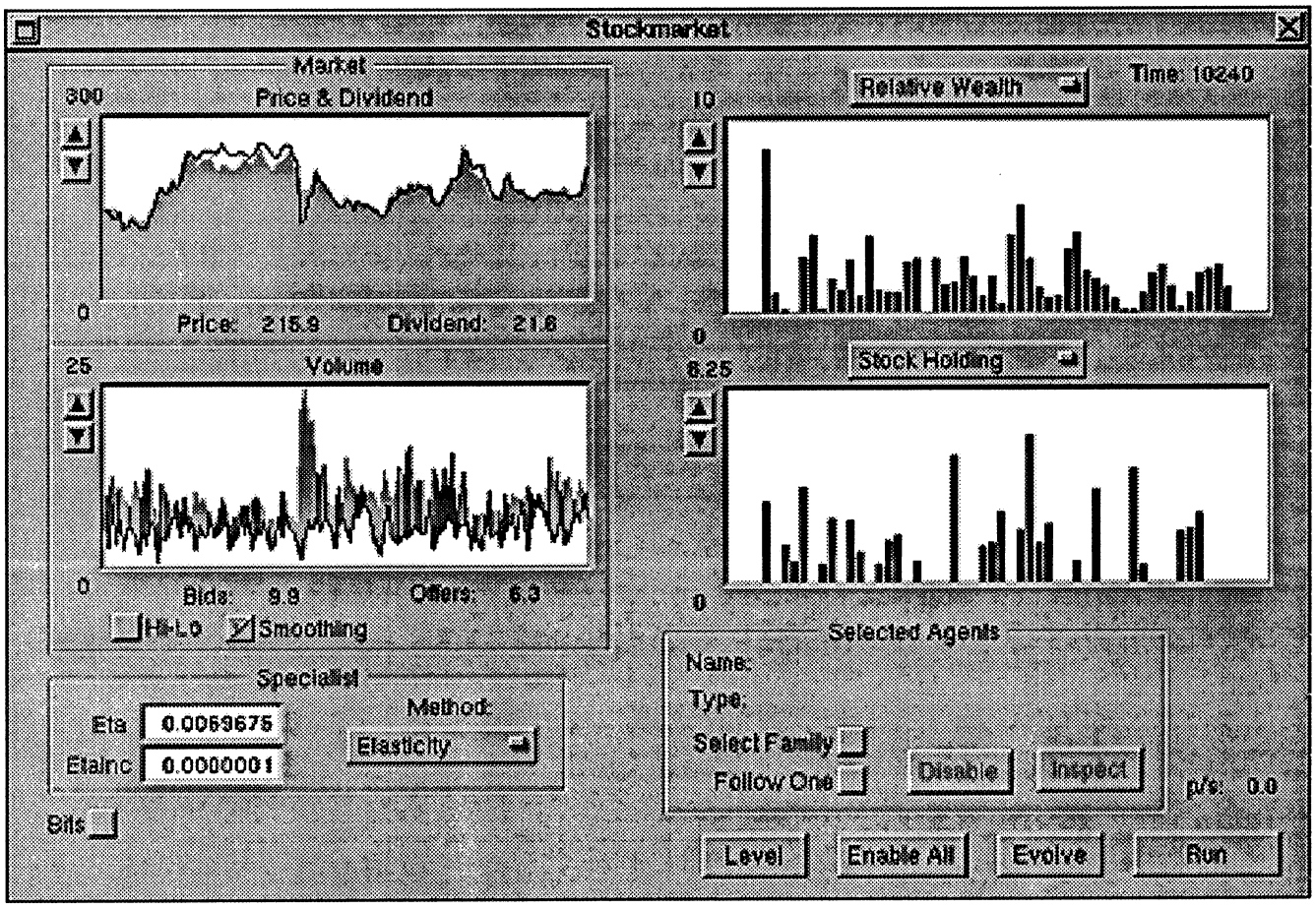


Figure 1. A frozen moment in the surrogate stock market.

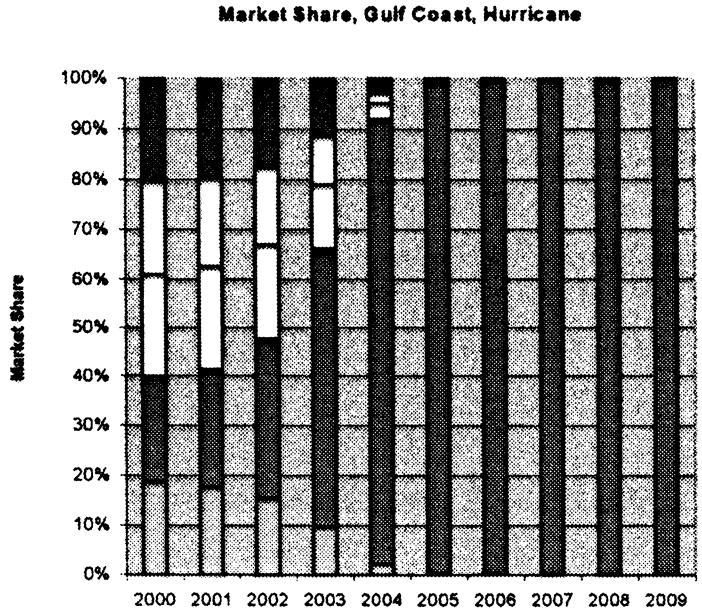


Figure 2. Market share distribution for five primary insurers.

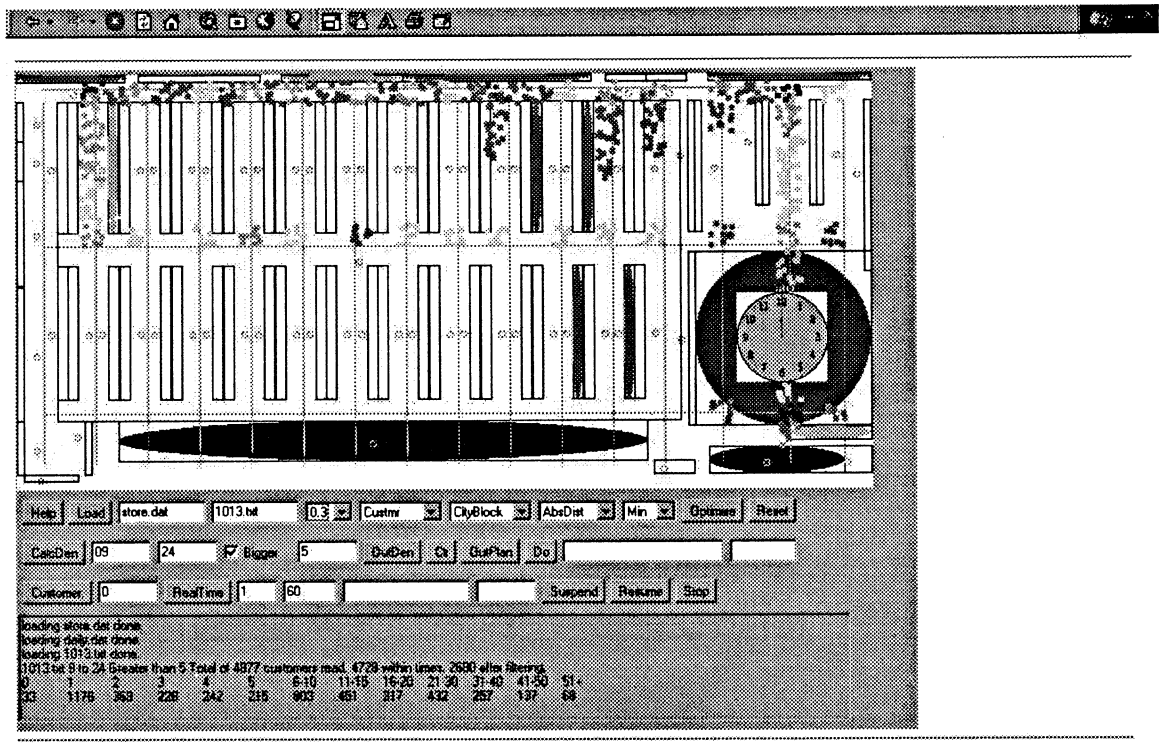


Figure 3. Customer densities along each aisle in the simulated store.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A UNIVERSAL ORIGIN OF LIFE

Steen Rasmussen* & Stirling Colgate*, Klaus Lachner*, Bruce Lehnert*,
Luigi Luisi+,Bernd Mayer#,Johndale Solem* and David Whitten*
(*)Los Alamos National Laboratory, (&)Santa Fe Institute
(+)ETHZ Zurich, (#) University of Vienna.

Abstract

We present a new picture of a possibly universal origin of life, which differs from the current paradigms where life is either believed to emerge in tidal pools at the planetary surface or near hydrothermal vents at the ocean floor.

We propose a sequence of fundamental processes of mainly chemical and astrophysical nature leading to the origin of life that then becomes a natural consequence of downhill thermodynamic processes. This work deals with the mechanisms for the formation of life beneath a (our) planetary surface, and the origin of the likely and necessary physical and chemical pre-conditions. From this sequence we have chosen two topics that we believe are the least understood.

(I) We focus on the minimum molecular assembly which can self-reproduce and which can support a template directed self-replicating molecule as our definition of life. We believe micelles or liposomes to be likely substrate and polypeptides as the most likely templating molecules. We have developed a nine-step process where some of the steps already have been verified experimentally and where experimental procedures are proposed to test the "missing links". Simple calculations show that all the critical molecules can be produced from known reaction paths within the vast pore space of the planetary crust consisting of water, hydrocarbons and minerals. The main energy source for a proto-organism consisting of a self-reproducing "shell" and a self-replicating "gene" is of interfacial nature. For the proto-organism to start harness chemical energy bound in the non-equilibrium mineral environment (forgassity) an additional cooperative structure has to be established. This same forgassity of the rocks, e.g. Fe3+/Fe2+, is used by some of the contemporary life forms at the root of the phylogenetic tree, the Acheae. The cooperative structure could be established by a self-assembling redox complex (possibly a polypeptide aggregate with metal irons) also residing within with the liposomemembrane. Perhaps this proposed path will lead to a new way to create life, but the boundary conditions should be the same this time and it includes a coupling of a self-reproducing liposome (proto-cell compartment), self-replicating polymers in the interface (proto-genes) and a redox complex (proto-metabolism).

(II) In addition we focus on the origin of the pre-planetary accretion disk which leads to the formation of the planets. By arguing that the optimum location for the origin of life is within the pore space in the rock beneath the crusted surface, this establishes the likely raw materials for the start of life and consequently the feasible pathways. It also raises the question of the critical processes leading up to this environment such as the mass of the earth, that of other planets, and how they are formed. We therefor look at the least understood of the astrophysical processes leading up to these conditions. This is the accretion disk mechanism and how it leads to the range of masses of the planets. We explore the consequences of this process for star and planet formation and in particular investigate the turn-off condition that leads to planet masses.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Evolving Robot Morphology and Control

Craig Mautner
Computer Science and Engineering
University of California, San Diego
La Jolla, CA. 92093-0114

Richard K. Belew
Computer Science and Engineering
University of California, San Diego
La Jolla, CA. 92093-0114

Abstract

Most robotic approaches begin with a fixed robot hardware design and then experiment with control structures. We take a different approach that considers both the robot hardware and control structure as variables in the evolutionary process. This paper reports the results of experiments which explore the placement of sensors and effectors around the perimeter of a simulated agent's body, and the Neural Network (NNets) that controls them.

1 Introduction

Evolutionary algorithms have been used in the design of robot controllers for some time and with great success. Almost all work on evolving autonomous agents and robots has focused on evolution of the control structure, often a NNet. This work has taken as a basic assumption that the agent body is immutable. This is a consequence of the flexibility of software over that of hardware: in the past it has not been feasible to explore the space of robot morphology. In contrast to this, the course of natural evolution shows a history of body, nervous system and environment all evolving simultaneously in cooperation with and in response to each other. The research proposed below investigates the interaction between co-evolved body and control structures.

The most successful agents we know of are those found in real life. These agents are well adapted to their environment and can handle many small and large surprises to their world and themselves without failures. Because of this we look to biology for much of our inspiration for this work. However, in contrast to some biorobotic models, we are not trying to reflect the results of our work back to the biology that inspires it.

The methods by which we investigate this is to specify the agent's body and NNet using a grammar. Grammars offer modularity in terms of encompassing detailed structures at various levels of granularity. That is, a grammar can provide a compact representation for complicated and repeated structures. By using grammars we are able to build hierarchical solutions to problems based on the solutions found at lower levels. These grammars are then evolved using common Genetic Algorithm techniques based on the performance of the instantiated agent.

Many aspects of this research have been investigated in isolation by others. Examples of evolved robots that have implemented NNet controllers include six legged walking controllers (Whitley [3], Kodjabachian [8]), maze following (Floreano [5]), predator-prey behavior (Floreano [6]), and food tracking (Angeline [1]).

An early experiment applying grammatical models to the construction of feed-forward NNets is due to Kitano [7]. Other researchers who have used grammars to develop NNets are Whitley and Gruau [3] and Lucas [10].

Very little prior work in evolving morphology exists. Menczer and Belew [12] investigated the nature of sensor usage by providing their agents with an evolvable NNet connected to sensors and effectors. Marks, Polani, and Uthmann [11] explored eye types and positioning. Sims [13] demonstrated a simulation where the complete morphology of the individuals was involved. Sims created an artificial world in which each agent was grown from a genome that defined both the physical structure and the control structure. Eggenberger [4] has developed an evolutionary system that simulates the growth of a body based on differential gene expression. Lee et. al. [9] have also worked on evolving both control structures and body plans.

2 Background

We have carried out several experiments based on the paradigms listed above. The experiments begin with a population of genome strings which are converted to individual grammars. The grammars are used to generate agent bodies and their associated NNet control

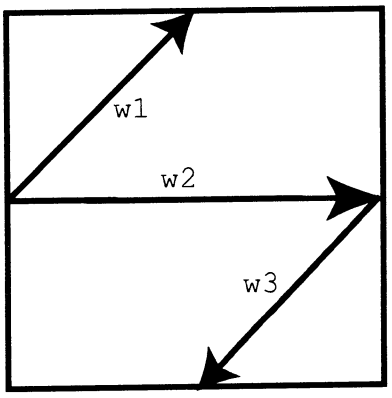


Figure 1: Terminal Cell Example

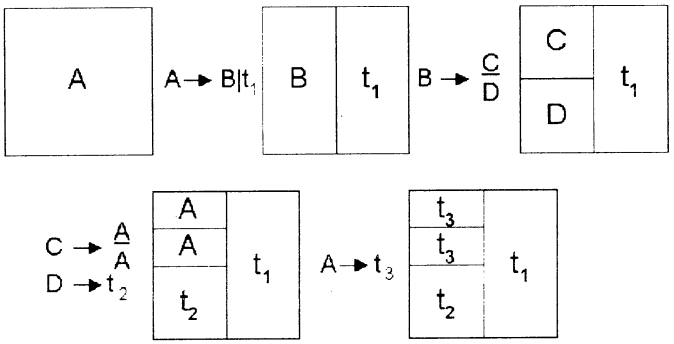


Figure 2: Development Process

systems. Through a development process that transforms a single undivided cell (the gamete) into a body consisting of several interconnected cells.

The body/controller pair is then evaluated in the context of an environment and the genomes for the more successful individuals are preferentially selected for reproduction in the next generation.

The process of cell division, neural network extraction, and fitness evaluation are described in the next sections.

2.1 Production Rules

The grammar consists of an alphabet of terminal and non-terminal symbols, a set of production rules, and a single starting symbol from the set of non-terminals. In our experiments the terminals are the hexadecimal characters {0-9a-f}, and the non-terminals are a subset of the uppercase characters {A-Z}.

Each production rule is made up of a left side and a right side. The left side is a single non-terminal symbol; the right side is a specification for the creation of one or two cells. A non-terminal may appear as the left side of more than one production rule.

The specification of each cell defines it to be either a single non-terminal symbol or a terminal cell which is a list of directed, weighted edges from the sides of the cell to each other (See Figure 1). The production rule also contains the orientation of the cell (whether it is horizontally or vertically flipped). Specifying a cell to be a non-terminal allows it to be subject to the application of further production rules.

A production rule also specifies whether the non-terminal produces one or two cells, and if two, the relative position (i.e. above, below, to the left of, or to the right of) of each.

2.2 Cell Division

As described in the previous section, each cell is either labeled with a non-terminal symbol or is a terminal cell. Initally the gamete is labeled with the starting non-terminal symbol of the grammar and cell differentiation proceeds by selecting and applying the rules of the grammar. For each cell labeled with a non-terminal, a rule is found whose left side matches the non-terminal. The cell is then replaced with the one or two cells specified by the right side of the rule. If there is no matching rule, the cell is replaced with a terminal cell with random weights and edges.

The process continues replacing non-terminals with terminals and non-terminals until there are only terminals left or a maximum depth of replacements have occurred. The rules are applied only a limited number of times to keep rules of the form A→A from generating an infinite regress. In our experiments, the maximum number of cell divisions was set to 6. This permits a body to have at most 64 cells (one cell divided in half 6 times produces $2^6 = 64$ cells). A derivation that continues beyond the sixth rule application will replace the cell by a terminal cell with random weights and edges.

A derivation that takes four generations is shown in Figure 2. In this example the gamete is labeled with the starting symbol, A. The production rule A→B|t_1 indicates that the non-terminal A is converted into two cells. The first cell, a non-terminal B, is to be placed to the left of the second which is the terminal cell, t_1.

2.3 Neural Network Interpretation

Once the cell division is complete, the body consists of a set of cells that have within them directed, weighted edges. The cells and edges are interpreted as sensors, effectors and the neural networks that connect them.

Each side of a cell has associated with it two values,

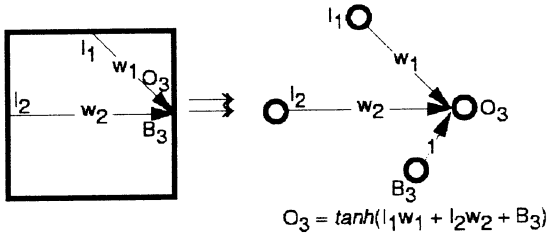


Figure 3: NNet edges formed from within cell connections

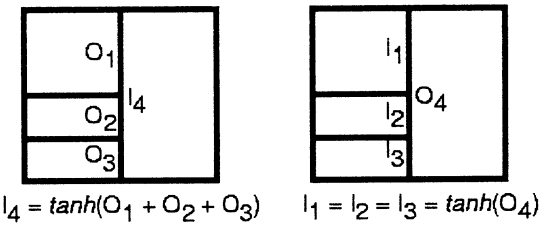


Figure 4: NNet edges formed from connections between adjacent cells

an input value and an output value. The input value is dependent on the output values of all adjacent cells. The output value of a side is dependent on the input value of all sides and the internal weighted edges between the sides. Both the input and output values are described in detail below.

2.3.1 Network Edges Within a Cell

There are three types of edges that can exist within the cell: Corner, Cross, and Tonic. Corner edges run from one side to an adjacent side. Cross edges run from a side to the opposite side and Tonic edges are interpreted as bias weights. The three types are shown in Figure 3. The edge from I_1 is of corner type with weight w_1, I_2 is of cross type with weight w_2, and T_3 is of type tonic providing activation of value T_3.

The output of a cell is the squashed sum of the weighted inputs to that cell. The weighting is only done across those edges that are defined for that cell. In Figure 3 there are two edges and one tonic defined for the cell pictured. The output of the cell on the right edge is the squashed sum of the products $I_1 w_1$, $I_2 w_2$ and the Tonic weight T_3. For all of our experiments the squashing function is the hyperbolic tangent function: $\tanh(x)$.

2.3.2 Edges Between Cells

When one or more cells are adjacent to the side of a cell, the outputs of the incident cell are combined to form the inputs to the adjacent cell. This is demonstrated in Figure 4.

In this case the inputs labeled I1, I2, and I3 are adjacent to the output labeled O4. Similarly, the input labeled I4 is adjacent to the outputs labeled O1, O2 and O3. A neural network layer is formed by setting the input value on an edge of a cell equal to the squashed sum of the outputs of the cells adjacent to that edge.

Figures 7, 9, 10 and 11 illustrate some final agents' terminal cell divisions and the neural networks that result. In these diagrams the semicircles pointing outwards are effectors and those semicircles pointing inwards are sensors.

2.3.3 Sensors and Effectors

Sensors and effectors are defined by the edges of the terminal cells. Any directed edge that originates from a cell side that is on the perimeter of the body becomes a sensor or input node. Any directed edge that terminates on a cell side that is on the perimeter of the body becomes an effector or output node.

Sensor nodes detect signals of the environment. They provide the input that is propagated through the NNet of the body. Effector nodes are the outputs of the NNet.

Effector nodes provide propulsion to the agent's body. The force of this propulsion is proportional to their output activation. The direction of propulsion of the agent is a result of summing all of these forces based on their position on the body. The vector sum of all effector outputs is broken into two forces: The first is a pressure, which acts through the center of the body and translates the body through the environment. The second force is torque which acts perpindicular to a line through the center of the body and causes the body to rotate. The net effect is demonstrated in Figure 5.

2.4 Application in a Simple Environment

In order to judge the effectiveness of a given body plan, and hence the effectiveness of the genome from which the body plan developed, the mature agent is placed within an environment. For our experiments the environment consists of a 500x500 world that has at its center a source of reward which produces a de-

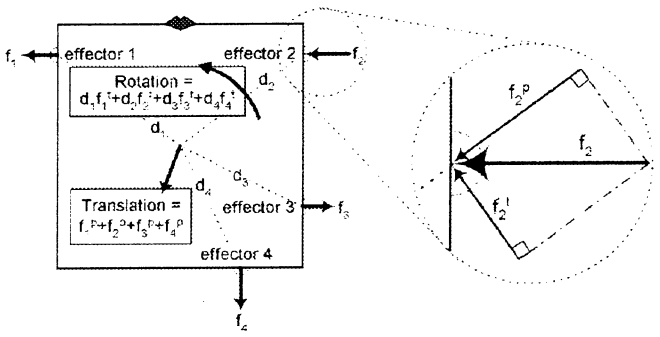


Figure 5: Conversion of Effector Outputs to Rotation and Translation

tectable signal that falls off as the inverse-square of the distance.

All agents' bodies are set to be 20 units on each edge. The body has a mouth placed at its top. The fitness is measured from the mouth. The perfect agent would turn it's mouth toward the center of the world and approach the center in as few time steps as possible.

The agents are placed at a random location in the world and then allowed to wander freely (or to just sit in the case of many agents) for 30 time steps. After 30 time steps the agent is moved to another random location. This is repeated for 10 placements for a total of 300 time steps. All agents in a given generation are started from the same set of locations. If an agent gets within a body length of the center of the world then it is moved the next location.

The fitness, f_i, at time step i is shown below where d is the distance from the agent's mouth to the center of the world.

$$f_i = \begin{cases} \frac{1}{d^2} & \text{if } d \geq \text{agent body length} \\ 1 & \text{otherwise} \end{cases}$$

The fitness, F, of an individual is the sum of the fitnesses at each time step over its lifetime.

2.5 The Braitenberg Vehicle

A classic design in the field of robotic control is the Braitenberg Vehicle 2b described in [2]. This agent has two sensors on the front and two effectors on the back. The agent's body is bi-laterally symmetric with each sensor connected via a positive weight to the effector on the opposite side. The effect of this connectivity is to steer the agent to the side with the stronger sensor. The effectiveness of this design has been demonstrated in a number of robots.

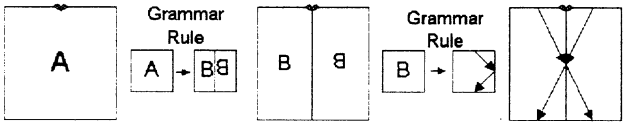


Figure 6: Grammar for Generating Braitenberg Vehicle

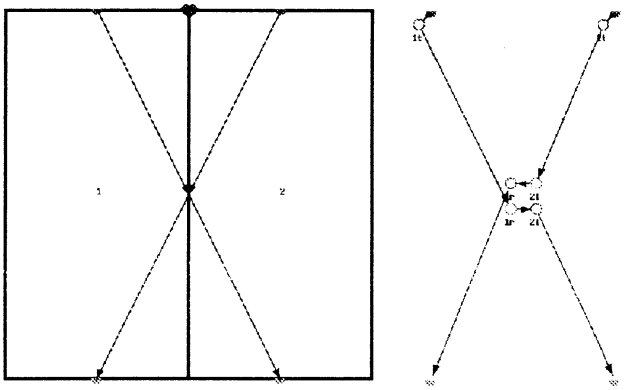


Figure 7: Braitenberg Vehicle with Cell and NNet Representations

A simple grammar that generates a complete Braitenberg body is shown in Figure 6. The grammar consists of two rules. The first rule rewrites the undifferentiated cell body (start symbol A) into two nonterminal B cells one of which is horizontally flipped relative to the other. The second rule converts a B nonterminal cells into a terminal cell with two edges defined. Figure 7 shows Braitenberg's Vehicle 2b alongside the one generated by this hand-designed grammar. The parsimonious nature of the grammar that generates the Braitenberg Vehicle under our system shows the representational adequacy of the grammar system.

2.6 Preliminary Results

Using the experimental platform described above we ran several experiments to determine the effectiveness of our approach. Three such results are described in the graph of Figure 8. The results show the average of the best agent over ten different runs with the same starting conditions. The first experiment is our hand-designed Braitenberg agent which is run in the environment for 100 generations. Its performance is shown by the short line around a fitness of 26. This experiment provides a baseline with which we may compare other results.

The top line of the graph begins with the same Braitenberg agents of the first experiment, but in this

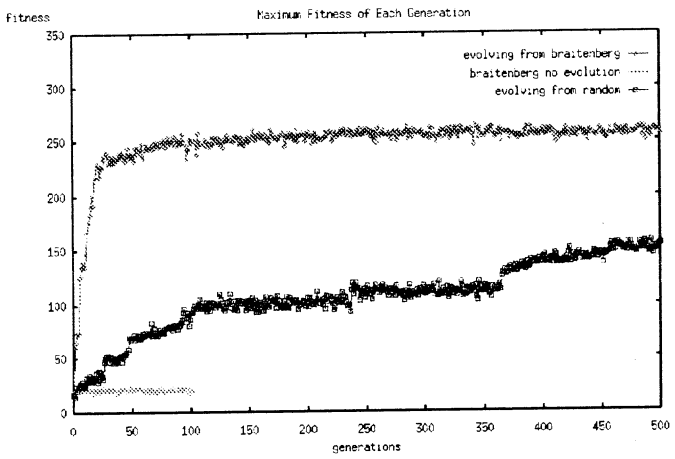


Figure 8: Average of Fitnesses (Best of Each Generation)

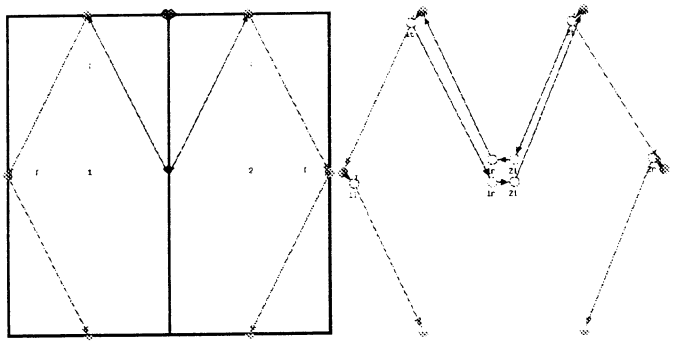


Figure 9: Agent and NNet evolved from Braitenberg

experiment they are allowed to evolve. Their performance shows significant improvement when compared to the original run of Braitenberg agents. All ten populations converged by the end of the run, although they each converged on different solutions. One commonality of all solutions was the preservation of the original *bi-lateral symmetry* that was part of the Braitenberg population prototype. However in about half of the final populations the connections were positive and crossed as in the Braitenberg but, surprisingly, the remaining populations converged on solutions that had negatively weighted connections straight down the body rather than across it. Most of the solutions also added two negative bias weights to the front of the agent which acted as tractor effectors pulling the body forward constantly. This left new sensors and effectors on the side free to steer the body towards the goal. A typical solution is shown in figure 9.

The final experiment (middle line) initialized the

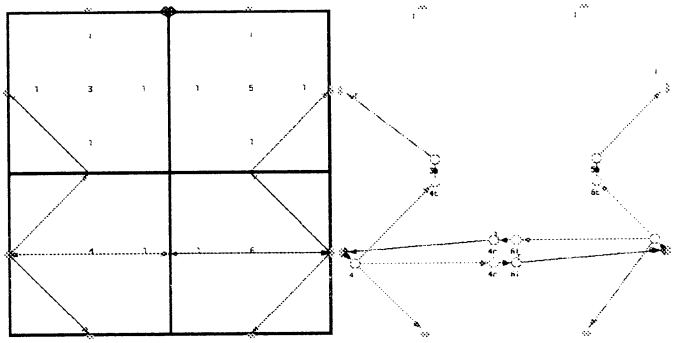


Figure 10: Agent and NNet evolved from Random Genome(Ex. 1)

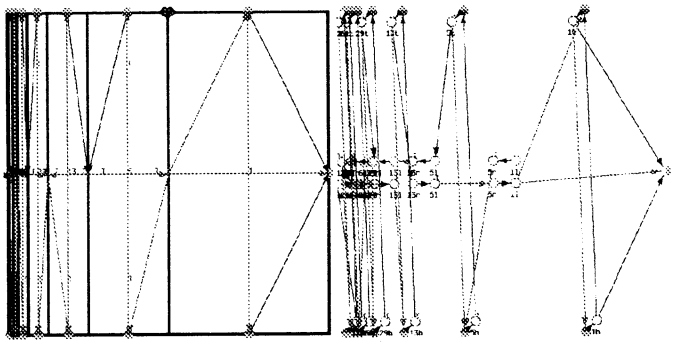


Figure 11: Agent and NNet evolved from Random Genome (Ex. 2)

population with random genomes. In this case results varied widely. Some runs were able to find steadily improving solutions within a few hundred generations while others took quite a bit longer. Some solutions found were Braitenberg-like in that they discovered bi-lateral symmetry (figure 10). Other solutions produced bizarre body types that performed very well (figure 11).

3 Summary

The conclusion that we can draw from these results is that even in this very simple world a wide range of potential solutions exist. Many of these solutions are superior to the known good *a priori* Braitenberg Vehicle solution. It was shown that we can evolve agents whose body and control structures are tightly coupled in order to solve the simple environment presented here. The fact that the system found so many solutions is encouraging as we apply it to more complex environments.

Our current work involves applying these tech-

niques to real robots including Kheperas and LEGObots whose body shapes and sensor and effector placement will be determined by the evolved grammars.

References

[1] P. Angeline, G. Saunders, and J. Pollack. An evolutionary algorithm that constructs recurrent networks. *IEEE Trans. on Neural Networks*, 5:54–65, 1994.

[2] V. Braitenberg. *Vehicles: Experiments in Synthetic Psychology.* MIT Press, Cambridge, 1984.

[3] F. Gruau D. Whitley and L. Pyeatt. Cellular encoding applied to neurocontrol. In L. Eshelman, editor, *International Conference on Genetic Algorithms.* Morgan Kaufmann, 1995.

[4] P. Eggenberger. Evolving morphologies of simulated 3d organisms based on differential gene expression. In Phil Husbands and Inman Harvey, editors, *Fourth European Conference on Artificial Life*, pages 205–213. MIT Press, 1997.

[5] D. Floreano and F. Mondada. Automatic creation of an autonomous agent: Genetic evolution of a neural network driven robot. In J.-A. Meyer D. Cliff, P. Husbands and S. Wilson, editors, *From Animals to Animats III*, Cambridge, MA, 1994. MIT Press.

[6] D. Floreano and S. Nolfi. Adaptive behavior in competing co-evolving species. In Phil Husbands and Inman Harvey, editors, *Fourth European Conference on Artificial Life.* MIT Press, 1997.

[7] H. Kitano. Designing neural networks using genetic algorithms with graph generation system. *Complex Systems*, 4(4), 1990.

[8] J. Kodjabachian and J. A. Meyer. Evolution and development of neural networks controlling locomotion, gradient following, and obstacle avoidance in artificial insects. http://www.biologie.ens.fr/fr/animatlab/perso/-kodjaba/kodjaba.html, 1997.

[9] W. Lee, J.C. Hallam, and H.H. Lund. Hybrid gp/ga approach for co-evolving controllers and robot bodies to achieve fitness-specified tasks. In *Proceeding of IEEE 3rd International Conference on Evolutionary Programming*, pages 384–389, New York, 1996. IEEE Press.

[10] S. M. Lucas. Growing adaptive neural networks with graph grammars. In *Proceedings of European Symposium on Artificial Neural Networks (ESANN '95)*, pages 235–240, 1995. http://esewww.essex.ac.uk/ sml/papers.html.

[11] Alexandra Mark, Daniel Polani, and Thomas Uthmann. A framework for sensor evolution in a population of braitenberg vehicle-like agents. In Christoph Adami, Richard K. Belew, Hiroaki Kitano, and Charles E. Taylor, editors, *Artificial Life VI*, pages 428–432, Cambridge, Mass, 1998. MIT Press.

[12] F. Menczer and R. K. Belew. Evolving sensors in environments of controlled complexity. In R. Brooks, editor, *Proc. Fourth Conf. on Artificial Life*, 1994.

[13] K. Sims. Evolving 3d morphology and behavior by competition. In R. Brooks and P. Maes, editors, *Proceedings of the International Conference Artificial Life IV*, Cambridge MA, 1994. MIT Press.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Grasping Impact Force Control of A Flexible Robotic Gripper Using Piezoelectric Actuator

W. Yim W.R. Wells

Dept. of Mechanical Eng.
University of Nevada, Las Vegas
Las Vegas, Nevada 89154-4027

Abstract

This paper presents a robust force tracking control of a flexible beam during grasping operation using piezoceramic actuator. Equations describing the motion of the gripper in condition of contact and noncontact are derived based on the cantilever beam. In this study, contact force is regulated in addition to the impact force generated at the instant of contact based on variable structure model reference adaptive control theory using only force measurement. For the derivation of the control law, it is assumed that parameters of the beam and object stiffness are unknown. It is shown in computer simulation that in the closed-loop system designed using bounds on uncertain functions, the contact force tracks a given reference trajectory and has a good transient behavior in spite of the imact force generated during the initial contact.

1 Introduction

A grasping device is very important for a large number of applications such as a robotic hand and considerable research has been given to the force control of the robotic gripper [1,4]. In this paper, a cantilever beam actuated by piezoelectric actuator is considered for a miniaturized robotic gripper.

Often the impact phenomenon during the initial stage of grasping is overlooked in contact force control. A real control task for a robotic manipulator implies, in general, several transitions between the condition of free motion and the condition of constrained motion. It implies the generation of undesired reaction forces at each sudden change from one condition to another. Conventional controller for robot manipulators has assumed either a free space motion or constrained motion in contact with a certain environment. Considerable efforts have been given to model the sudden change of the motion equations that happens when the robots and external environement switch sharply from a condition of noncontact to a condition of contact.[3,7]

Research on smart structure systems using piezoelectric materials was undertaken by many researchers. Bailey and Hubbard [8] used piezoceramic actuator to achieve active vibration damping for a cantilever beam using a distributed-parameter control theory. Other researches include: study of stiffness effects of piezoelectric actuators on elastic properties of the host structures [10], vibration control through a modal shape analysis [2], force tracking control of flexible gripper [5,9]. Experimental results on force tracking control of a cantilever beam using piezoelectric actuator are reported by Choi et al [1]. Especially in [9], a sliding mode controller is utilized for force tracking control of a two- fingered flexible gripper based on the truncated model of the cantilever beam. Force control problem with a flexible gripper becomes complicated due to the interaction between the control and structural flexibility of the beam, where smart materials are attached on.

The contribution of this paper lies in the design of a control system for the control of contact force of a cantilever beam based on the theory of variable structure model reference adaptive control (VS-MRAC) [13-14], using only input and output signals. The input signal is the voltage applied to piezoelectric actuator and the output signal is the output voltage of the force sensor attached at the tip of the cantilever beam. Unlike the published works in literature [2,9], both linear and nonlinear dynamic terms of the cantilever beam, the stiffness of the contact surface as well as a sudden change of system dynamics due to impact phenomenon are assumed to be unknown to the designer. However it is assumed that an upper bound on the uncertain functions is given or can be estimated. It is shown by simulation that in the closed-loop system, contact force tracks given reference trajectory and beam vi-

bration is suppressed in spite of dynamic parameter variations.

2 Dynamic Model

In this Section the dynamic model of a cantilever beam which has a piezoelectric actuator bonded on the top surface as shown in Fig.1 is derived. The object surface is modeled as a spring and located d from the tip of the cantilever beam. When a control voltage u(x,t) is applied to the piezoelectric actuator, the induced strain ϵ_p in the piezoceramic is given by [8]

$$\epsilon_p(x,t) = u(x,t)\frac{\alpha}{h_p} \tag{1}$$

where α is the static piezoelectric constant, and h_p is the thickness of piezoceramic layer. Based on [8], the bending moment, M_p, generated by the input voltage $u(x,t)$ is given by

$$M_p(x,t) = cu(x,t) \tag{2}$$

where c is a constant determined by a given beam material and geometry as well as piezoceramic and expresses the bending moment per volt. The c can be a function of x when the material properties and geometry of the beam change along x.

By considering the cantilever beam shown in Fig.1 as the Bernoulli-Euler beam, the kinetic energy T and potential energy V can be expressed as

$$\begin{aligned} T &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} m_{ij}\dot{q}_i(t)\dot{q}_j(t) \qquad (3)\\ V &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} k_{ij}\dot{q}_i(t)\dot{q}_j(t) + \int_0^l \sum_{i=1}^{N}(\phi_i'' q_i(t))cu(t)dx \\ &\quad + \frac{1}{2EI}\int_0^l c^2u^2(t)dx + \frac{1}{2}k_s(\sum_{i=1}^{N}\phi_i(l)q_i(t) - d)^2 \end{aligned}$$

where $m_{ij} = \int_0^l \rho A\phi_i\phi_j dx$, $k_{ij} = \int_0^l EI\phi_i''\phi_j'' dx$ and k_s is the stiffness of contact surface. $EI = E_1I_1 + E_2I_2$, $\rho = \rho_1 + \rho_2$, l is the length of the beam, and $A = A_1 + A_2$ are the effective bending stiffness, density, and cross-sectional area of the cantilever beam and the piezoceramic respectively. A subscript $()_1$ refers a cantilever beam and $()_2$ refers a piezoceramic layer. The mode shape function $\phi_i(x)$ is from the boundary condition of cantilever beam and beam deflection $y(x,t)$ is approximated as

$$y(x,t) = \sum_{i=1}^{N}\phi_i(x)q_i(t) \tag{4}$$

to obtain a finite dimensional model.

Applying Lagrange equation and augmenting damping term, the following equations of motion can be obtained:

$$M\ddot{q}(t) + C\dot{q}(t) + Kq(t) + K_s(q(t)) = b_u u(t) \tag{5}$$

where M and K are constant mass matrix and stiffness matrices whose elements are m_{ij} and k_{ij} respectively. It should be noted that $K_s = 0$ for $y(l,t) \le d$. The C is a diagonal damping matrix and $K_s = k_s[(\sum\phi_i(l)q_i - d)\phi_1(l), (\sum\phi_i(l)q_i - d)\phi_2(l), \ldots]^T$ and $b_u = -c[\int\phi_1', \int\phi_2', \ldots]^T$.

3 Variable Structure Adaptive Control

In this section, the variable structure adaptive control is derived for contact force control for $d = 0$. [19] Defining the state vector $x = (q^T, \dot{q}^T)^T$, the system (5) can be written in a state variable form as

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 0 & U \\ -M^{-1}K & -M^{-1}C \end{bmatrix} x + \begin{bmatrix} 0 \\ M^{-1}b_u \end{bmatrix} u \\ &\quad - \begin{bmatrix} 0 \\ M^{-1}K_s(q) \end{bmatrix} \qquad (6)\\ &= Ax + Bu - E(x) \end{aligned}$$

and the contact force at the tip of the cantilever beam is defined as the output of the system as

$$f = \begin{bmatrix} R & 0 \end{bmatrix} x \triangleq h^T x \tag{7}$$

where $R = k_s[\phi_1(l)\ \phi_2(l)\ \ldots]$ and 0 and U are a null and identity matrices of appropriate dimensions.

Consider the input-output representation of the system (7) given by

$$\begin{aligned} f &= h^T(sU - A)^{-1}(Bu - E(x)) \qquad (8)\\ &= W(s)[u + (-W^{-1}(s)h^T(sU - A)^{-1}E(x))] \\ &= W(s)[u + g(x,t)] \end{aligned}$$

where

$$\begin{aligned} W(s) &= h^T(sU - A)^{-1}B \triangleq \frac{k_p n_p(s)}{d_p(s)} \\ g(x,t) &= -W^{-1}(s)h^T(sU - A)^{-1}E(x) \end{aligned}$$

where n_p, d_p are appropriate monic polynomials, $g(x,t)$ is a configuration dependent function, and s denotes the Laplace variable or the differential operator. Here we are interested in the solution of (7) in a bounded subset Ω of the state space $\Re^{2N}$.

Let f_m be a smooth reference trajectory generated by a reference model. We are interested in deriving an

output feedback control law $u(t)$ such that the contact force tracking error $e = f - f_m$ asymptotically tends to zero and elastic modes remain bounded during the motion in spite of the sudden transition of system dynamics from free to constrained motion. It should be noted that for the design of the control system, the parameters of the cantilever beam and piezoelectric actuator, i.e. M, C, K, K_s, b_u, the stiffness k_s of the contact surfac, the initial distance of object d are unknown.

Since the relative degree of the system is 2 and minimum phase, a reference model of relative degree 2 with input r and the output f_m is considered.

$$\begin{aligned} f_m &\triangleq W_m(s)r \qquad (9)\\ W_m(s) &= \frac{k_m}{s^2+\alpha_{m1}s+\alpha_{m2}} \triangleq \frac{k_m}{d_m(s)} \end{aligned}$$

where the poles of W_m are assumed to be stable. Now the control law will be derived for tracking the reference force trajectory f_m. Consider a controllable and observable representation of (9) given by

$$\begin{aligned} \dot{\zeta} &= A_1\zeta + b(u+g(x,t)) \qquad (10)\\ f &= h_1^T\zeta \end{aligned}$$

where $W(s) = h_1^T(sU - A_1)^{-1}b$.

For the synthesis of the controller, define the regressor vector $\omega = [\omega_1^T, f, \omega_2^T, u]^T \in \Re^{4N}$ where ω_1, ω_2 are the output of the following filters:

$$\begin{aligned} \dot{\omega_1} &= \Lambda\omega_1 + \nu u \qquad (11)\\ \dot{\omega_2} &= \Lambda\omega_2 + \nu f \end{aligned}$$

where $\omega_1, \omega_2 \in \Re^{2N-1}, \nu \in \Re^{2N-1}$,

$$\begin{aligned} \Lambda &= \begin{bmatrix} -\lambda_{2N-2} & -\lambda_{2N-3} & \cdots & -\lambda_o \\ & U & & 0 \end{bmatrix} \qquad (12)\\ \nu &= \begin{bmatrix} 1 & 0 \end{bmatrix}^T \end{aligned}$$

and λ_i are coefficients of the polynomial $det(sU-\Lambda) = s^{2N-1} + \lambda_{2N-2}s^{2N-2} + \ldots + \lambda_1 s + \lambda_o$.

Under the assumption of $sgn(k_p) = sgn(k_m)$ and $g(x,t) = 0$, it is known [12] that there exists a unique constant vector $\theta^* = (\theta_1^{*T}, \theta_f^*, \theta_2^{*T}, \theta_r^*) \in \Re^{4N}$ such that the transfer function of the closed-loop system with $u = \theta^{*T}\omega$ matches $W_m(s)$ exactly, i.e. $f = W(s)u = W(s)\theta^{*T}\omega = W_m(s)r$.

The system (11) and filters (12) can be written in compact form by defining the vector $X^T = (\zeta^T, \omega_1^T, \omega_2^T)^T \in \Re^{6N-2}$ as

$$\begin{aligned} \dot{X} &= \begin{bmatrix} A_1 & 0 & 0 \\ 0 & \Lambda & 0 \\ \nu h_1^T & 0 & \Lambda \end{bmatrix} X + \begin{bmatrix} b \\ \nu \\ 0 \end{bmatrix} u + \begin{bmatrix} b \\ 0 \\ 0 \end{bmatrix} g \qquad (13)\\ &\triangleq A_o X + B_o u + B_1 g \end{aligned}$$

and

$$\begin{aligned} f &= \begin{bmatrix} h_1^T & 0 \end{bmatrix} X \qquad (14)\\ &= h_o^T X \end{aligned}$$

Let $\tilde{u} = u - u^* = u - \theta^{*T}\omega$, and $\kappa^* = k_p/k_m$ and

$$u^* = \begin{bmatrix} \theta_1^{*T} & \theta_f^* & \theta_2^{*T} \end{bmatrix} DX + \theta_r^* r \qquad (15)$$

where

$$D = \begin{bmatrix} 0 & U & 0 \\ h_1^T & 0 & 0 \\ 0 & 0 & U \end{bmatrix} \qquad (16)$$

By adding and substracting $B_o u^*$ to the right hand side of the equation (13) and using (15) for u^*, eqs (13) and (14) can be written as

$$\begin{aligned} \dot{X} &= A_c X + B_c\kappa^*\tilde{u} + B_c r + B_1 g \qquad (17)\\ f &= h_o^T X \end{aligned}$$

where $A_c = A_o + B_o(\begin{bmatrix} \theta_1^{*T} & \theta_f^* & \theta_2^{*T} \end{bmatrix} D)$, $B_c = B_o\theta_r^*$ and $B_o = B_c\kappa^*$ since $\kappa^* = k_p/k_m = 1/\theta_r^*$. For $u = u^*$ and $g = 0$, (17) represents the reference model. Thus $W_m(s) = h_o^T(sU - A_c)^{-1}B_c$ and system (22) gives

$$f = W_m(s)r + \kappa^* W_m(s)\tilde{u} + g_c \qquad (18)$$

where $g_c = \hat{W}_m(s)g$, $\hat{W}_m(s) = h_o^T(sU - A_c)^{-1}B_1$. Here $W_m(s)$ is a stable function and g_c is bounded for any bounded function g.

Considering the reference model (9) as

$$\begin{aligned} \dot{X}_m &= A_c X_m + B_c r \qquad (19)\\ f_m &= h_o^T X_m \end{aligned}$$

and defining the state error $e = X - X_m$, one has the following error equation

$$\begin{aligned} \dot{e} &= A_c e + B_c\kappa^*\tilde{u} + B_1 g \qquad (20)\\ e_o &= h_o^T e \end{aligned}$$

From (25) the output tracking error becomes

$$e_o = \kappa^* W_m\tilde{u} + g_c \qquad (21)$$

The output error equation (21) plays an important role in the derivation of control law. For the synthesis of the controller, it is essential to introduce a chain of auxilliary errors (e_i'). Since the relative degree of n^* of the reference model is two, a polynomial $L(s)$ of a degree $(n^* - 1)$ is chosen so that $W_m(s)L(s)$ is strictly positive real (SPR). For our cantilever beam force controller, $L(s)$ of the form

$$L(s) = \frac{s+\delta}{\delta} \qquad (22)$$

is chosen and $\delta > 0$. A chain of auxiliary errors e_i' $(i = 0, 1)$ are generated by the following set of filtered signals:

$$\chi_1 = u \quad (23)$$

$$\chi_o = L^{-1}\chi_1 \quad (24)$$

$$\xi_1 = \omega \quad (25)$$

$$\xi_o = L^{-1}\xi_1 \quad (26)$$

It should be noted that auxiliary error $e_i'(i = 0, 1)$ are governed by an SPR transfer function, namely, $W_m(s)L(s)$ for e_o' and $L^{-1}(s)$ for e_1', and a modulation functions μ_i, $i = 0, 1$. Complete alogorithm is shown in Fig.2

In Fig.2, θ_{nom} and κ_{nom} are nominal values of the parameters θ^* and κ^* obtained from some nominal model of the plant. Note that y_a can be interpreted as a predicted output error and, hence, e_o' is a prediction error. The transfer functions $W_m(s)L(s)$ and $L^{-1}(s)$ are associated with e_o' and e_1'. It should be noted that each error e_i' is governed by a SPR transfer function. Computation of modulation function μ_i can be done on-line using the available signal ξ_i and χ_i using (24) and (26). However a significant simplication in control law synthesis can be obtained by using constant modulation function by the suitable choice of μ_i. A practical way of selecting the modulations functions $\mu_i(i = 0, 1)$ is by overestimated values of the bound g in the computation of μ_i and adjust them accordingly based on the closed-loop system responses. It should be noted that this simplification is obtained at the expense of larger modulation functions.[15]

The term $(u_o)_{eq}$ in Fig.2 is the equivalent control, which is well described in the variable structure system literature [18]. The signal $(u_o)_{eq}$ is obtained from u_o by means of a low-pass filter with high enough cut-off frequency dominated averaging filter [16]. Following [13, 14], the VS-MRAC shown in Fig.2 has the following properties.

- The errors $e_i'(i = 0, 1)$ all converge to zero in finite time.
- The contact force tracking error $(f - f_m)$ converges exponentially to zero.

4 Simulation Results

In this section, simulation results for the closed-loop system with the control law are presented using Matlab and Simulink. Mechanical properties of the simulated cantilever beam with piezoelectric actuator is shown in Table 1.

	beam	piezoceramic
density(kg/m^3)	2000	8000
Young's Modulus(GPa)	8	64
width (mm)	30	30
height (mm)	1	1
length (mm)	200	200
Piezo-strain constant(m/volt)		300E-12

Table 1: Mechanical properties of cantilever beam

The step response of the uncontrolled system with initial conditions of $f(0) = \dot{f}(0) = 0$ with $d = 0.05m$ is shown in Fig. 3 and 4. Fig. 3 shows oscillatory responses. Now the simulation results for variable structure adaptive control is presented with initial conditions of $f(0) = \dot{f}(0) = 0$. In the proposed controller, a simplified relay type controller was synthesized by using the constant modulation functions. The values of μ_i are chosen in several trials by observing simulated responses. The constant values chosen in this simulation were $\mu_1 = 50E6$ and $\mu_2 = 10E07$. The reference model is chosen as

$$W_m(s) = \frac{100}{s^2 + 20s + 100} \quad (27)$$

and $L(s)$ is chosen as

$$L^{-1}(s) = \frac{10}{s + 10} \quad (28)$$

We note that $W_m(s)L(s)$ is SPR.

For the computation of $u_{nom} = \theta_{nom}^T\omega$, the nominal value of θ^* were arbitrarily chosen as $\theta_{nom} = (0, \ldots, 1)^T$. In the saturation function, the boundary layer thickness was set to $\epsilon = 0.1$. The parameter of the averaging filter was chosen as $\tau = 10E - 6$ after several trials by observing simulated responses.

Simulations were performed to examine the effect of uncertainty in contact surface stiffness on controller performance for a contact force of $f^* = 0.1N$. Input r of Fig.2 was set to $r = f^*$. The initial conditions were assumed to be $\dot{q}(0) = q(0) = 0$. The controller parameters were tuned by observing simulated responses for the truncated model (6) for three modes. Two different values of stiffness were tested, i.e. $20000N/m$ and $200N/m$. Fig.5 and 6 show the force tracking and tip deflection plots for two different cases. Insensitivity of the proposed adaptive controller to uncertainty in stiffness parameter k_s is apparent from Fig. 5 and 6. Fig.7 is the plot of control voltages applied to the piezoelectric actuator for force tracking control of Fig. 5 and 6.

Simulation was also done to check insensitivity of the proposed controller for uncertain surface locations.

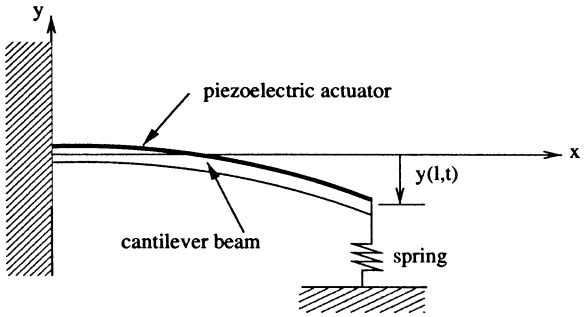


Figure 1: Schematic diagram of a cantilever beam

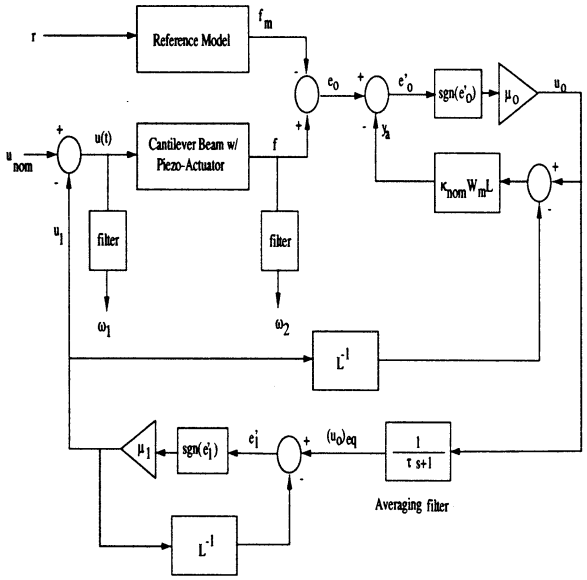


Figure 2: Controller block diagram

Fig. 8 shows the plot of contact force trajectory for the surface stiffness of $20000N/m$.

5 Conclusions

Based on the variable structure model reference adaptive control theory, a control law for the force control of the cantilever beam is considered for smooth transition during impact phase. In the cantilever beam model, system parameters and the stiffness of the surface as well as the location are assumed unknown. A variable structure adaptive control law was synthesized using only measurement of contact force. In the closed-loop system, contact force tracked the reference model output trajectory and smooth regulation of beam vibration were accomplished.

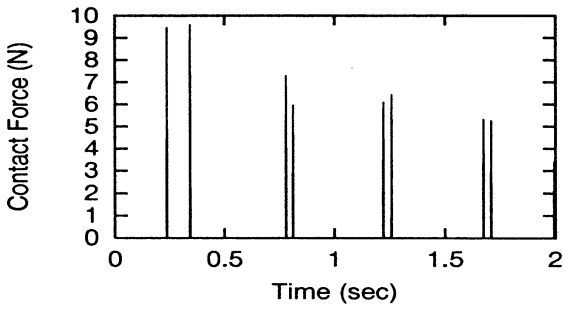


Figure 3: Open loop response for contact force

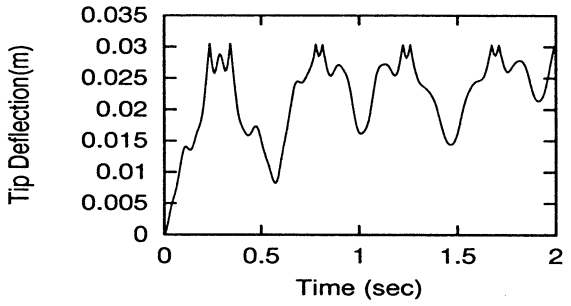


Figure 4: Open loop response for tip deflection

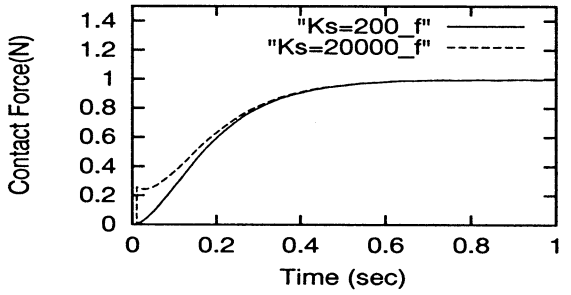


Figure 5: Effects of surface property change on contact force

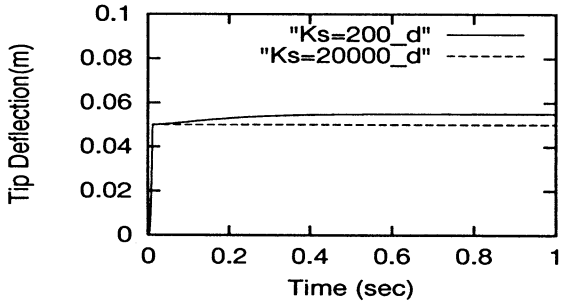


Figure 6: Effect of surface property change on tip deflection

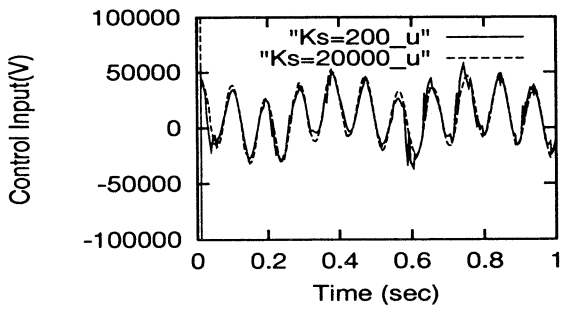


Figure 7: Plot of control input

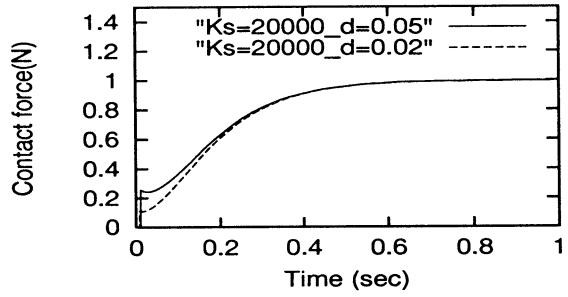


Figure 8: Effect of initial impact distance

References

[1] Choi, Seung-Bok, Cheong, Chae-Cheon, and Lee, Chul-Hee, "Position tracking control of a smart flexible structure featuring a piezofilm actuator," Journal of Guidance, Control and Dynamics, Vol 19, No 6, Nov-Dec, 1996.

[2] Chonan, S., Jiang, Z.W. and Sakuma, S.,"Force control of a miniature gripper driven by piezoelectric bimorph cells," Trans. of the JSME(Series C), Vol 59, No 557, pp 150-157.

[3] Tornambe, A., " Global regulation of a planar robot arm striking a surface," IEEE Trans. on Automatic Control, vol. 41, No. 10, Oct. 1996.

[4] Mills, J. K. and Lokhorst, D. M., "Control of robotic manipulators during general task execution: A discontinuous control approach," The Int Journal of Robotic Research, vol. 12, No 2, pp. 146-163, 1993.

[5]Mills, J. K. and Nguyen, C. V., "Robotic manipulator collisions: Modeling and Simulation," Trans of ASME, J. of Dynamic Systems, Measurement, and Control, Vol 114, Dec. 1992, pp 650-659.

[6] Volpe, R. and Khosla, P., "A theoretical and experimental investigation of impact control for manipulators," Int. Journal of Robotic Research, vol. 12, No. 4, pp 351-365, 1993.

[7] Zheng, Y. F. and Hemami, H. "Mathematical modeling of a robot collision with its environment," Journal of Robotic Systems, vol. 2, No. 3, pp 289-307, 1985.

[8] Bailey, Thomas and Hubbard, James E., "Distributed piezoelectric-polymer active vibration control of a cantilever beam," Journal of Guidance, Control and Dynamics, Vol 8, No 5, Sept-Oct, 1985, pp 605-611.

[9] Choi, Seung-Bok, Lee, Chul-Hee, "Force tracking control of a flexible gripper driven by piezoceramic actuators," ASME Transaction Journal of Dynamic Systems, Measurement, and Control, Vol 119, September 1997.

[10] Baz, A. and Poh, S., "Performance of an active control system with piezoelectric actuator," Journal of Sounds and Vibration, Vol 126, No 2, pp 327-343, 1988.

[11] Utkin, V.I.,"Sliding modes and their applications in variable structure system", MIR, 1978.

[12] Narendra, K.S. and Annaswamy, A.,"Stable adaptive systems", Prentice Hall, 1989.

[13] Hsu, L, Araujo, A.D., and Lizarralde, F, "New results on input/output variable structure model reference adaptive control: Design and stability analysis", IEEE Transaction on Automatic Control, Vol 42, March 1997, pp 386- 393.

[14] Hsu, L., Araujo, A.D., and Costa, R.R., "Analysis and design of I/O based variable structure adaptive system", IEEE Transaction on Automatic Control, Vol AC-39, No 1, 1994, pp 4-21.

[15] Hsu, L, "Variable structure model reference adaptive control using only input and output measurements: The general case," IEEE Transactions on Automatic Control, Vol AC-35, No 11, 1990, pp 1238-1243.

[16] Araujo, A.D., and Singh, S., "Variable structure adaptive control of wing-rock motion of slender delta wings," Journal of Guidance, Control and Dynamics, March-April, 1998

[17] Ioannon, P.A. and Sun, J., Robust Adaptive Control, Prentice Hall PTR., 1996.

[18] Utkin, V.I., Sliding Modes and Their Applications in Variable Structure Systems, MIR, Moscow, 1978.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

LEARNING ALGORITHMS IN A CLASS OF KNOWLEDGE-BASED SYSTEMS

Z. Bubnicki

Institute of Control and Systems Engineering,
Wroclaw University of Technology,
Wyb. Wyspianskiego 27, 50 – 370 Wroclaw, POLAND
phone: +48 71 320 33 28 , +48 71 21 62 26 ; fax: +48 71 320 38 84
email: bubnicki@ists.pwr.wroc.pl

Keywords: learning algorithms, knowledge-based systems, logical systems

Abstract

The control system with a static plant described by the knowledge representation in the form of relations and in the form of logical formulas is considered. The learning process consists in using the successive knowledge validation and updating to the determination of the current control decisions. Two approaches and algorithms have been described: for the validation and updating of the knowledge on the plant and on the form of the control. In both cases two versions are presented: the learning process in open-loop and in closed-loop control system. For the plant with the logical knowledge representation the logic-algebraic method is applied.

1. Introduction. Decision making problem

A great variety of definitions and approaches in the field of learning control systems has been described (see e.g. [10,11,12]). The purpose of this paper is to present a review of concepts and algorithms for a class of knowledge-based systems with a static plant described by a knowledge representation in the form of relations or a set of facts (logical knowledge representation). Let us consider a plant described by a set of relations

$$R_i(x, y, z, c) \subset X \times Y \times Z \ , \qquad i = 1, 2, \ldots, m \quad (1)$$

where $x \in X$, $y \in Y$, $z \in Z$ are input, output and external disturbance vectors, respectively, and c is the vector of parameters in the relations R_i defined in $X \times Y \times Z$.

The relations (1) may be defined by a set of inequalities. For the result of observations in the form $z \in D_z$ where $D_z \subset Z$, the set (1) may be reduced to one relation $R(x, y, c) \subset X \times Y$:

$$R(x, y, c) = \{(x, y) \in X \times Y : \bigvee_{z \in D_z} (x, y, z) \in \bigcap_{i \in \overline{1,k}} R_i\}. \quad (2)$$

E.g. for the plant with k inputs, one output and one parameter

$$(x^{(1)})^2 + (x^{(2)})^2 + \ldots + (x^{(k)})^2 + y^2 \le c^2 \quad (3)$$

where $x = [\, x^{(1)} \ \ldots \ x^{(k)} \,]^T$. The decision making or **control problem** may be formulated as follows: For the given (1), the result of the observations D_z and the set D_y (user's requirement) one should find the largest set $D_x(c) \subset X$ such that the implication $x \in D_x(c) \rightarrow y \in D_y$ is satisfied. It is easy to see that

$$D_x(c) = \{ x \in X : \ D_y(x) \subseteq D_y \} \quad (4)$$

where

$$D_y(x) = \{ y \in Y : \ (x, y) \in R(x, y, c) \}. \quad (5)$$

The condition for D_y such that $D_x(c) \neq \varnothing$ (empty set) may be considered as the controllability condition. In the papers [2,3,5] the logical formulas as specific forms of R_i and the properties $x \in D_x$, $y \in D_y$ have been considered, and the first concepts of learning consisting in the current knowledge evaluation and updating for the plants with unknown parameters have been described [4,6,8]. The new cases, results and generalisations presented in this paper are divided into two parts concerning the knowledge on the plant (i.e. the relation

the relation R) and the knowledge on the control (i.e. the set D_x). The idea of learning presented in the paper may be considered as a generalisation of the known concept of the adaptive control using the results of identification (e.g. [1]). For the further consideration we assume that $R(x, y, c)$ is a continuous and closed domain in $X \times Y$. In Sec.4 the considerations for the logical knowledge representation are presented.

2. Validation and updating of the knowledge on the plant

Version A — Off-line knowledge validation and updating

Assume now that the parameter c in the relation describing the plant has the value $c = \bar{c}$ and $\bar{c}$ is unknown. If we have the sequence of observations

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \ , \bigwedge_i [(x_i, y_i) \in R(x, y, \bar{c})]$$

then we can propose an estimation of the unknown value $\bar{c}$. In the second part of this section the current step by step estimation will be described. On each step one should prove if the current observation "belongs" to the knowledge representation determined to this step (**knowledge validation**) and if not – one should modify the current estimation of parameters in the knowledge representation (**knowledge updating**).

Let us introduce the set

$$D_c(n) = \{ c \in C : \bigwedge_i [(x_i, y_i) \in R(x, y, c)] \} . \quad (6)$$

It is easy to see that $D_c(n)$ is a closed set in C. The boundary $\Delta_c(n)$ of the set $D_c(n)$ is proposed here as the estimation of $\bar{c}$. It is easy to note that at least one point (x_i, y_i) belongs to $\Delta_c(n)$. In the example (3) the set $D_c(n) = [c_{\min}, \infty)$ and $\Delta_c(n) = \{c_{\min}\}$ where

$$c_{\min}^2 = \max_i (x_i^{\mathrm{T}} x_i + y_i^2) . \quad (7)$$

Assume that the points (x_i, y_i) occur randomly from $R(x, y, \bar{c})$ with probability density $f(x, y)$, i.e. that (x_i, y_i) are the values of random variables $(\underline{x}, \underline{y})$ with probability density $f(x, y)$. Then the following theorem concerning the convergence of $\Delta_c(n)$ may be proved:

Theorem 1

If $f(x, y) > 0$ *for each* $(x, y) \in R(x, y, \bar{c})$ *and for each* $c \neq \bar{c}$ $R(x, y, c) \neq R(x, y, \bar{c})$ *then* $\Delta_c(n)$ *converges to* $\{\bar{c}\}$ *with probability* 1.

The idea of the determination of $\Delta_c(n)$ may be presented in the form of the following recursive algorithm for $n > 1$.

Knowledge validation

One should prove if

$$\bigwedge_{c \in D_c(n-1)} [(x_n, y_n) \in R(x, y, c)] . \quad (8)$$

If yes then $D_c(n) = D_c(n-1)$ and $\Delta_c(n) = \Delta_c(n-1)$. If not then one should determine the new $D_c(n)$ and $\Delta_c(n)$, i.e. update the knowledge:

Knowledge updating

$$D_c(n) = \{ c \in D_c(n-1) : (x_n, y_n) \in R(x, y, c) \} \quad (9)$$

and $\Delta_c(n)$ is the boundary of $D_c(n)$.

For $n = 1$

$$D_c(1) = \{ c \in C : (x_1, y_1) \in R(x, y, c) \} . \quad (10)$$

The control is based on the result of learning, i.e. for the final value n the parameter c_n is chosen randomly from $\Delta_c(n)$ and the decision x_n is selected randomly from $D_x(c_n)$.

Version B — Learning process in the closed-loop system

The idea of the closed-loop learning system presented here consists in the following: (1) For the successive decision x_n based on the current knowledge of the plant and its result y_n, the knowledge validation and updating should be performed. (2) Then the next decision x_{n+1} is based on the updated knowledge. To complete the procedure it is necessary to determine how to choose the value c_n from $\Delta_c(n)$ (if $\Delta_c(n)$ contains more than one element) and how to choose the control decision x_n from the set $D_x(c_n)$. They may be chosen randomly with the fixed probability distribution for $\Delta_c(n)$ and probability density $f(x)$ for $D_x(c_n)$. If $D_x(c_n) = \varnothing$ (the controllability condition is not satisfied for $\bar{c} = c_n$) then x_n is chosen randomly from X without the restriction to $D_x(c_n)$. Finally, the **control algorithm** with the knowledge validation and updating in the closed-loop learning system is the following:

1. Put x_n at the input of the plant and measure y_n at the output.
2. Prove the condition (8) (knowledge validation).
3. Determine $D_c(n)$ and $\Delta_c(n)$. If (8) is not satisfied, the knowledge updating according to (9) is necessary.
4. Choose randomly c_n from $\Delta_c(n)$.
5. Determine $D_x(c_n)$ according to (4) with $\bar{c} = c_n$.
6. Choose randomly x_{n+1} from $D_x(c_n)$.
7. If $D_x(c_n) = \varnothing$, choose randomly x_{n+1} from X.

For $n = 1$, choose randomly x_1 from X and determine $D_c(1)$ in (10).

The block scheme of the learning control system is presented on Fig.1. For the random choice of c_n and x_n the generators G_1 and G_2 of the random numbers are required. Their probability distributions are precised currently for $\Delta_c(n)$ and $D_x(c_n)$.

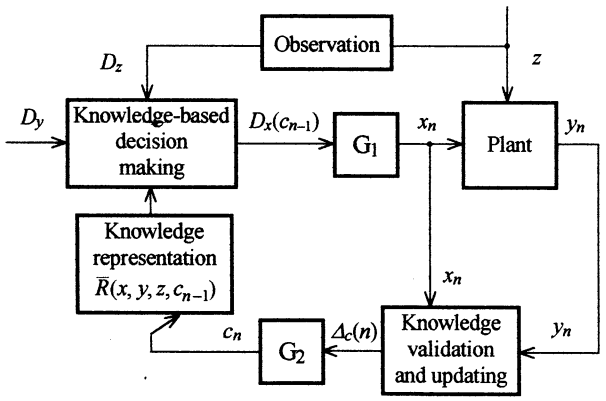


Fig. 1

Example: Consider very simple example with one-input and one-output plant described by the relation

$$c\,z\,x \le y \le d\,z\,x \;,\quad d > c > 0 \quad (\bar{c}^{(1)} = c,\ \bar{c}^{(2)} = d).$$

For $D_z = [z_1, z_2]$ where $z_2 > z_1 > 0$, the relation (2) has the form $c\,z_1\,x \le y \le d\,z_2\,x$. For the requirement $D_y = [y_{\min}, y_{\max}]$, $y_{\min}, y_{\max} > 0$ – it is easy to note that

$$D_x(c, d) = \left[\frac{y_{\min}}{c z_1}, \frac{y_{\max}}{d z_2}\right] \qquad (11)$$

and the controllability condition is: $d\,z_2\,y_{\min} < c\,z_1\,y_{\max}$. The set (6) is determined by

$$c \le \frac{1}{z_1}\min_i \frac{y_i}{x_i} \;,\quad d \ge \frac{1}{z_2}\max_i \frac{y_i}{x_i}$$

and $\Delta_c(n)$ may be reduced to

$$c_n = \frac{1}{z_1}\min_i \frac{y_i}{x_i} \;,\quad d_n = \frac{1}{z_2}\max_i \frac{y_i}{x_i}\;.$$

In our example x_n is chosen according to the rectangular probability density`

$$f(x) = \begin{cases} \dfrac{1}{\beta - \alpha} & \text{for } \alpha \le x \le \beta \\ 0 & \text{otherwise} \end{cases}\;.$$

Taking into account (11) we can assume $x_n > 0$ and $X = [0, \bar{\beta}]$ with $\bar{\beta}$ sufficiently great. The control algorithm in the closed-loop system is then the following:

1. Put x_n at the input and measure y_n.
2. Knowledge validation and updating:

if $c_{n-1}\,z_1\,x_n \le y_n \le d_{n-1}\,z_2\,x_n$ then $c_n = c_{n-1}$, $d_n = d_{n-1}$

if $y_n < c_{n-1}\,z_1\,x_n$ then $c_n = \dfrac{y_n}{z_1 x_n}$, $d_n = d_{n-1}$

if $y_n > d_{n-1}\,z_2\,x_n$ then $c_n = c_{n-1}$, $d_n = \dfrac{y_n}{z_2 x_n}$.

3. According to (11) determine

$$x_{\min,n} = \frac{y_{\min}}{z_1 c_n} \;,\quad x_{\max,n} = \frac{y_{\max}}{z_2 d_n}\;.$$

4. If $x_{\min,n} < x_{\max,n}$ – choose randomly x_{n+1} for $\alpha = x_{\min,n}$, $\beta = x_{\max,n}$.
5. If $x_{\min,n} > x_{\max,n}$ – choose randomly x_{n+1} for $\alpha = 0$, $\beta = \bar{\beta}$ where $\bar{\beta} >> \dfrac{y_{\max}}{z_2 d}$.

The control and learning process has been simulated and investigated for the different probability distributions.

3. Validation and updating of the knowledge on the control

In this approach the validation and updating concerns directly $D_x(c)$, i.e. the knowledge on the form of control. When the parameter $\bar{c}$ is unknown then for the fixed value of x it is not known if x is a correct decision, i.e. if $x \in D_x(\bar{c})$ which implies $y \in D_y$. Our problem may be considered as a classification or pattern recognition problem with two classes. The point x should be classified to class $j = 1$ if $x \in D_x(\bar{c})$ and to class $j = 2$ if $x \notin D_x(\bar{c})$. Assume that we can use the learning sequence

$$(x_1, j_1),\ (x_2, j_2),\ \ldots,\ (x_n, j_n) \triangleq S_n \qquad (12)$$

where $j_i = \{1,2\}$ are the results of the correct classification given by a trainer (a teacher) for a sequence of points $x_1, x_2, \ldots, x_n$. The learning sequence (12) may be used to the current estimation of $\bar{c}$ and consequently to the current updating of the recognition algorithm determined by the form of $D_x(c)$. Two versions analogous to those presented in Sec.2 may be considered. Assume for the further considerations that $D_x(c)$ is a continuous and closed domain in X.

Version A — Off-line knowledge validation and updating

According to this approach two time intervals are introduced. In the first interval the learning sequence is obtained for the sequence of inputs x_i chosen randomly from X. In the second interval the control is based on the result of learning, i.e. x_i are chosen randomly from $D_x(c_n)$ where c_n is the value of $\bar{c}$ obtained at the end of learning. Let us denote by $\bar{x}_i$ the subsequence for which $j_i = 1$, i.e. $\bar{x}_i \in D_x(\bar{c})$ and by $\hat{x}_i$ the subsequence for which $j_i = 2$, and introduce the following sets in C:

$$\bar{D}_c(n) = \left\{c \in C : \bar{x}_i \in D_x(c) \text{ for each } \bar{x}_i \text{ in } S_n\right\}, \qquad (13)$$

$$\hat{D}_c(n) = \{c \in C : \hat{x}_i \in X - D_x(c) \text{ for each } \hat{x}_i \text{ in } S_n\}. \tag{14}$$

The set

$$\overline{D}_c(n) \cap \hat{D}_c(n) \triangleq \overline{\Delta}_c(n)$$

is proposed here as the estimation of $\bar{c}$. Assume that x_i is chosen randomly from X with probability density $f(x)$. Then the following theorem concerning the convergence of $\overline{\Delta}_c(n)$ may be proved:

Theorem 2

If $f(x) > 0$ *for each* $x \in X$ *and for each* $c \neq \bar{c}$ $D_x(c) \neq D_x(\bar{c})$ *then* $\overline{\Delta}_c(n)$ *converges to* $\{\bar{c}\}$ *with probability* 1.

For example, let $D_x(c)$ is described by inequality

$$(x^{(1)})^2 + (x^{(2)})^2 + \ldots + (x^{(k)})^2 \leq c^2 \tag{15}$$

where $x = [x^{(1)} \ldots x^{(k)}]^T$. Then

$$\overline{D}_c(n) = [c_{\min}, \infty) \quad , \quad \hat{D}_c(n) = [0, c_{\max}) \ ,$$

$$\overline{\Delta}_c(n) = [c_{\min}, c_{\max})$$

and $\overline{\Delta}_c(n) \to c$ w.p.1 , where $c_{\min}^2 = \max_i \bar{x}_i^T \bar{x}_i$,

$c_{\max}^2 = \min_i \hat{x}_i^T \hat{x}_i$.

The recursive **learning algorithm** for $n > 1$ is then the following:

If $j_n = 1$ $(x_n = \bar{x}_n)$

Prove if

$$\bigwedge_{c \in \overline{D}_c(n-1)} [x_n \in D_x(c)] .$$

If yes then $\overline{D}_c(n) = \overline{D}_c(n-1)$. If not – determine new $\overline{D}_c(n)$

$$\overline{D}_c(n) = \{c \in \overline{D}_c(n-1) : x_n \in D_x(c)\} .$$

Put $\hat{D}_c(n) = \hat{D}_c(n-1)$.

If $j_n = 2$ $(x_n = \hat{x}_n)$

Prove if

$$\bigwedge_{c \in \hat{D}_c(n-1)} [x_n \in X - D_x(c)] .$$

If yes then $\hat{D}_c(n) = \hat{D}_c(n-1)$. If not – determine new $\hat{D}_c(n)$.

$$\hat{D}_c(n) = \{c \in \hat{D}_c(n-1) : x_n \in X - D_x(c)\} .$$

Put $\overline{D}_c(n) = \overline{D}_c(n-1)$, $\overline{\Delta}_c(n) = \overline{D}_c(n) \cap \hat{D}_c(n)$.

For $n = 1$, if $x_1 = \bar{x}_1$ determine

$$\overline{D}_c(1) = \{c \in C : x_1 \in D_x(c)\} ,$$

if $x_1 = \hat{x}_1$ determine

$$\hat{D}_c(1) = \{c \in C : x_1 \in X - D_x(c)\} .$$

If for all $i \leq p$ $x_i = \bar{x}_i$ $(x_i = \hat{x}_i)$ – put $\overline{D}_p = X$ $(\hat{D}_p = \varnothing)$.

Version B — Learning process in the closed-loop system

The control decisions may be determined and put at the input of the plant currently during the learning process. In such a case the determination of the successive decision x_{n+1} is based on the estimation of $\bar{c}$ at the moment n. In the control algorithm proposed here the value c_n is chosen randomly from $\overline{\Delta}_c(n)$ and the value x_{n+1} is chosen randomly from $D_x(c_n)$ with the fixed probability distribution determined for $\overline{\Delta}_c(n)$ and $D_x(c_n)$, respectively. If $D_x(c_n) = \varnothing$ (the controllability condition is not satisfied for $\bar{c} = c_n$) then x_n is chosen randomly from X without the restriction $D_x(c_n)$. To know without a trainer if x_i is a correct decision, it is necessary to measure y_i and to prove if $y_i \in D_y$. The plant generating y_i together with the verification of the relation $y_i \in D_y$ may be considered as a trainer giving $j_i \in \{1,2\}$ for the successive values x_i in the learning sequence (12). This concept leads to the closed-loop control system (Fig.2) where G_1 and G_2 are the generators or the random numbers. The control algorithm is now the following:

1. Using the learning sequence S_n (12) determine $\overline{\Delta}_c(n)$ as it was described in the former algorithm.
2. Choose randomly c_n from $\overline{\Delta}_c(n)$.
3. Put $c = c_n$ in $D_x(c)$.
4. Choose randomly x_{n+1} from $D_x(c_n)$.
5. If $D_x(c_n) = \varnothing$, choose randomly x_{n+1} from X.

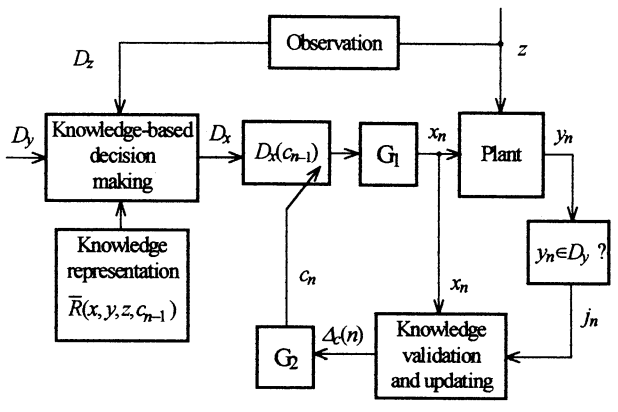


Fig.2

4. Logical knowledge representation

In many cases the knowledge representation given by an expert has a form of a set of relations

$$R_i(x, y, z, c) \quad , \quad i = 1, 2, \ldots, k$$

Then

$$R(x, y, c) = \{(x, y) \in X \times Y : \bigvee_{z \in D_z} (x, y, z) \in \bigcap_{i \in \overline{1,k}} R_i\}.$$

In this section we shall consider the knowledge representation in which the relations R_i have the form of logic formulas concerning x, y, z, c. Let us introduce the following notation for the plant under consideration:

1. $\alpha_{xi}(x,c)$ – simple formula (i.e. simple property) concerning x and c, $i = 1, 2, \ldots, n_1$, e.g. $\alpha_{x1}(x,c) =$ "$x^{\mathrm{T}}x \le c^{\mathrm{T}}c$".
2. $\alpha_{zr}(x,y,z,c)$ – simple formula concerning x, y, z and c, $r = 1, 2, \ldots, n_2$.
3. $\alpha_{ys}(y,c)$ – simple formula concerning y and c, $s = 1, 2, \ldots, n_3$.
4. $\alpha_x = (\alpha_{x1} \ldots \alpha_{xn_1}), \alpha_z = (\alpha_{z1} \ldots \alpha_{zn_2}), \alpha_y = (\alpha_{y1} \ldots \alpha_{yn_3})$.
5. $\alpha = (\alpha_x, \alpha_z, \alpha_y)$ – sequence of all simple formulas in the knowledge representation.
6. $F_j(\alpha)$ – j-th fact given by an expert. It is a logic formula composed with the subsequence of α and the logic operations: $\vee$ – **or**, $\wedge$ – **and**, $\neg$ – **not**, $\rightarrow$ – **if** ... **then**, $j = 1, 2, \ldots, k$. E.g. $F_1 = \alpha_1 \wedge \alpha_2 \rightarrow \alpha_4$, $F_2 = \alpha_3 \vee \alpha_2$ where $\alpha_1 =$ "$x^{\mathrm{T}}x < c^{\mathrm{T}}c$", $\alpha_2 =$ "the temperature is small or $y^{\mathrm{T}}y \le 3$", $\alpha_3 =$ "$y^{\mathrm{T}}y = z^{\mathrm{T}}x + c^{\mathrm{T}}c$", $\alpha_4 =$ "$y^{\mathrm{T}}y \ge 2c^{\mathrm{T}}c$".
7. $F(\alpha) = F_1(\alpha) \wedge F_2(\alpha) \wedge \ldots \wedge F_k(\alpha)$.
8. $F_x(\alpha_x)$ – input property, i.e. the logic formula using α_x.
9. $F_y(\alpha_y)$ – output property.
10. $F_z(\alpha_z)$ – property concerning z.
11. $a_m \in \{0, 1\}$ – logic value of α_m, $m = 1, 2, \ldots, n$, $n = n_1 + n_2 + n_3$.
12. $a = (a_1, a_2, \ldots, a_n)$.
13. $F(a)$ – the logic value of $F(\alpha)$. All facts given by an expert are assumed to be true, i.e. $F(a) = 1$.
14. $< \alpha, F(\alpha) > = KR$ (knowledge representation).

In this version of KR

$$R_i(x,y,w,c) = \{(x,y,z) \in X \times Y \times Z : F_i(a) = 1\}, \quad i \in \overline{1,k}$$

$$D_x(c) = \{ x \in X : F_x(a_x) = 1 \},$$

$$D_y(c) = \{ y \in Y : F_y(a_y) = 1 \}, \tag{16}$$

$$D_z = \{ z \in Z : F_z(a_z) = 1 \}.$$

The control problem analogous to that in Sec.1 is now formulated as follows: Given $F(\alpha)$, α_x, α_y, F_z and the required output property $F_y(\alpha_y)$ – find the best input property $F_x(\alpha_x)$ such that the implication $F_x \rightarrow F_y$ is satisfied. If it is satisfied for F_{x1} and F_{x2}, and $F_{x2} \rightarrow F_{x1}$ then F_{x1} is better than F_{x2}.

It may be shown that finding F_x is reduced to solving two sets of equations:

$$\left.\begin{aligned} F_z(a_z) \wedge F(a_x, a_z, a_y) &= 1 \\ F_y(a_y) &= 1 \end{aligned}\right\}$$

and (17)

$$\left.\begin{aligned} F_z(a_z) \wedge F(a_x, a_z, a_y) &= 1 \\ F_y(a_y) &= 0 \end{aligned}\right\}$$

with respect to a_x. If S_{x1} denotes the set of all solutions of the first equation (i.e. the set of all a_x for which there exists a_w, a_y such that $F_z \wedge F = 1$ and $F_y = 1$), and S_{x2} denotes the set of all solutions of the second equation – then F_x is determined by $S_x = S_{x1} - S_{x2}$. Given S_x, the formula F_x is determined in the known way, such that $a_x \in S_x \leftrightarrow F_x(a_x) = 1$. E.g. if $\alpha_x = (\alpha_{x1}, \alpha_{x2}, \alpha_{x3})$ and $S_x = \{(1,0,1), (0,1,1)\}$ then

$$F_x(\alpha_x) = (\alpha_{x1} \wedge \neg\alpha_{x2} \wedge \alpha_{x3}) \vee (\neg\alpha_{x1} \wedge \alpha_{x2} \wedge \alpha_{x3}).$$

It is worth to note that the solution of our problem is reduced to solving the algebraic equations (17) where F_z, F, F_y are the algebraic expressions in two-value logic algebra. It is just the main idea of the **logic-algebraic method**. The details, examples and the application of the decomposition to the problem solving may be found in [2,3,4,5,6,8].

5. Conclusions and final remarks

The learning process presented in the paper for a class of control systems consists in using the results of the successive knowledge validation and updating to the determination of the current control decisions. Two approaches and algorithms have been described: for the validation and updating of the knowledge on the plant and of the knowledge on the form of the control. The control decisions x and the values of the parameters c should be chosen randomly from the determined sets according to the given probability distributions $f(x)$ and $f_c(c)$. The simulations have shown the significant influence of the shape of these distributions on the convergence of the learning process. The comparison

between the different approaches requires further investigations.

In the papers [6,7,9] the new idea of so called uncertain variables has been introduced for the decision making in systems with unknown parameters. The modification of the presented learning algorithms using *a priori* information on c in the form of a certainty distribution given by an expert and describing the uncertain variable c may be interesting and promising.

References

[1] Bubnicki Z(1980), Identification of Control Plants. Elsevier; Oxford, Amsterdam, New York.

[2] Bubnicki Z(1997), Logic-algebraic method for a class of knowledge based systems. In: *Lecture Notes in Computer Science*, Springer Verlag; Berlin, pp. 420-428.

[3] Bubnicki Z(1997), Logic-algebraic approach to a class of knowledge based fuzzy control systems. *Proc. of European Control Conference*, Brussels, Vol.1, TU-E-G2.

[4] Bubnicki Z(1997), Knowledge updating in a class of knowledge-based learning control systems. *Systems Science*, Vol.23, No.4, pp. 19-36.

[5] Bubnicki Z(1998), Logic-algebraic method for knowledge-based relation systems. *Systems Analysis Modelling and Simulation*, Vol.33, pp.21-35.

[6] Bubnicki Z(1998), Uncertain variables and learning algorithms in knowledge-based control systems. *Artificial Life and Robotics*, Vol.3 (in press).

[7] Bubnicki Z(1998), Uncertain variables in knowledge-based systems. *Proc. of IASTED International Conference on Intelligent Systems and Control*, Halifax, Canada, pp. 135-139.

[8] Bubnicki Z(1998) Learning processes and logic-algebraic method in knowledge-based control systems. In: *Progress in System and Robot Analysis and Control Design*, Springer Verlag (to be published).

[9] Bubnicki Z(1998), Uncertain logics, variables and systems. *Proc. of the Third Workshop of the International Institute for General Systems Studies* (IIGSS), Beidaihe, Qinhuangdao, China, pp. 7-14.

[10] Gorinevsky D(1997), An approach to parametric nonlinear least square optimization and application to task-level learning control. *IEEE Trans. on Automatic Control*, Vol.42, No.7, pp. 912-927.

[11] Lucibello P(1998), A learning algorithm for improved hybrid force control of robot arms. *IEEE Trans. on Systems, Man and Cybernetics (Part A)*, Vol.28, No.2, pp. 241-244.

[12] Saab S.S, Vogt W.G, Mickle M.H(1997), Learning control algorithms for tracking 'slowly' varying trajectories. *IEEE Trans. on Systems, Man and Cybernetics (Part B)*, Vol.27, No.4, pp. 657-670.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Evolution of Differentiated Multi-threaded Digital Organisms

Thomas S. Ray & Joseph Hart
ATR Human Information Processing Research Laboratories
2-2 Hikaridai, Seika-cho Soraku-gun, Kyoto 619-02 Japan
ray@hip.atr.co.jp
jhart@hip.atr.co.jp

Abstract

Descriptive natural history of the results of evolution of differentiated multi-threaded (multi-cellular) self-replicating machine code programs (digital organisms), living in a network of computers, network Tierra. Programs are differentiated in that different threads execute different code (express different genes). The seed organism develops into a mature ten-celled form differentiated into a two-celled reproductive tissue and an eight-celled sensory tissue. The sensory threads obtain data about conditions on the machines in the network, and then process that data to choose the best machine to migrate to or to send the daughter to. Evolution leads to a diversity of algorithms for foraging for resources, primarily CPU time, on the network.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Robot populations and their controlled evolution

DJG James* and VG Rumchev**
* Control Theory and Applications Centre, Coventry University, Coventry, CV1 5FB, UK,
email: g.james@coventry.ac.uk

** School of Mathematics and Statistics, Curtin University of Technology, GPO Box U 1987, Perth, WA 6845, Australia, email: rumchevv@cs.curtin.edu.au

Abstract: In the paper recent results in controllability theory for a class of positive linear discrete-time systems are employed to examine and analyse reachability and controllability properties of cohort-type population models representing the dynamics of autonomous intelligent robot communities.

Key words: autonomous robots, positive systems, controllability, reachability

1. Introduction

In the robotics research literature, over recent years, significant interest has been shown in societies of autonomous intelligent robots or colonies of robots. Such multi-robot systems are seen as essential for undertaking tasks such as space station exploitation, planetary exploration and undersea exploration/ exploitation, in which they will be required to complete tasks without human controls. Clearly such robots will be required to have a degree of autonomy, implying an ability to interact with their environment, make necessary decisions and take appropriate actions including improving individual and collective performances. Aspects of multi-robot systems that have been reported on in the literature include: self organisation, especially when the society becomes large (ie hundreds of robots) (Sekiyama and Fukuda[1], Takadama et al[2]); social behaviour within the robot population (Agah and Bekey[3]); communication within societies and with other societies (Yoshida et al[4]); and interaction with humans (Suzuki et al[5]). The paper by Agah[7] provides a good review, including an extensive bibliography, of research in the field of distributed intelligent systems, including multi-robot systems. Reported research in Yoshida et al[4] also indicate that for proper diffusion of information and therefore effective communication within a society, essential for effective completion of group tasks, it is desirable to limit the number of robots within a society.

The field of autonomous systems is still in its infancy but there is the expectation of rapid progress over the next decade. In this paper the problem of robot population dynamics is considered under the assumption that they are capable, both by reinforced learning and by interacting with the environment, of continual self-evolution.

The robot population is categorised into generations or cohorts and its dynamics formulated as a cohort population model (Luenberger[8]); this being a specific discrete-time positive dynamic system with the duration of a single time period corresponding to the basic cohort span. Whilst the dynamics of cohort population models is well understood reachability and controllability properties, which are of fundamental importance, of such models have not been studied so far. In this paper recent results in controllability theory for positive linear discrete-time systems (Rumchev and James[9], Carretta and Rumchev[10]) are employed to examine and analyse reachability and controllability properties of some cohort-type population models as applied to intelligent robot populations.

2. Robot populations

A robot population is categorized into n generations (or cohorts) G_i , $i = 0,1,2,..., n-1$, with $x_i(t) \geq 0$, $t = 0,1,2,...$, denoting the number of robots of generation G_i at time period t, where a single time period corresponds to the time between the release of new generations of robots (say, for example, 2 years). Each new generation of robots exhibit incremental evolution in relation to the previous generation, with G_0 representing the newest generation (ie robots developed during the last time period $(t - 1, t)$) and G_{n-1} represents the oldest active generation, becoming obsolete at the end of the current time period $(t, t + 1)$. It is assumed that controlled migration, from other robot communities, to each generation is allowable, with

$$u_i(t) \geq 0 \ , \ i = 0,1,2 \ ,..., \ n - 1, \ t = 0,1,2,...$$

denoting the migration to generation G_i.

During one time period $(t - 1, t)$ the cohort constituting the ith generation simply moves up to the $(i + 1)$th generation, subject to attenuation due to some of the robots becoming non-functional and obsolete. Thus

$$x_{i+1}(t+1) = \beta_i x_i(t) + u_{i+1}(t), \quad \begin{cases} i = 0,1,2,\ldots,n-2 \\ t = 0,1,2,\ldots \end{cases} \tag{1}$$

where β_i , $0 \leq \beta_i \leq 1$ is the survival rate of the ith generation during one period.

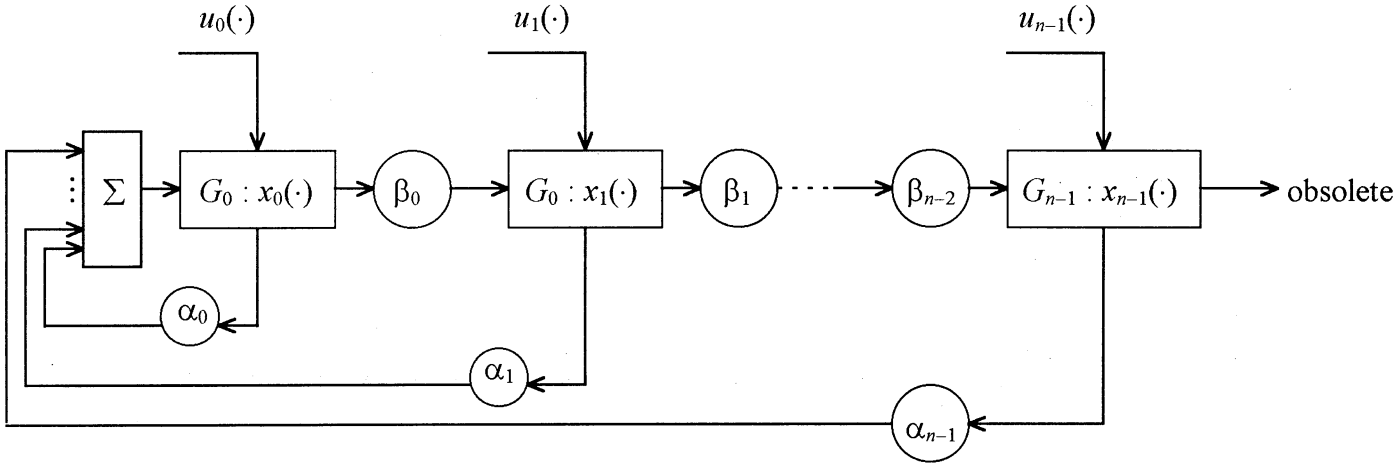


Fig 1: Robot Population dynamics model

It is assumed that, at least some, robots within each generation are able to contribute to the development of a new generation of robots. Under this assumption, and recognising the possibility of a controlled external inflow, the number $x_0(t+1)$ of new generation robots developed during the time interval $(t, t+1)$ is given by

$$x_0(t+1)=\alpha_0 x_0(t)+\alpha_1 x_1(t)+\ldots+\alpha_{n-1}x_{n-1}(t)+u_0(t),\quad t=0,1,2,.., \tag{2}$$

where $\alpha_i \geq 0$ is the production rate of the ith cohort of robots.

Together (1) and (2) constitute the discrete-time model for the robot population dynamics and may be represented diagramatically as in Fig 1.

3. Positive systems

Consider the open-loop positive linear discrete-time system (PLDS) (Luenberger[8], Szidarovsky and Bahill[11]) with vector control sequence

$$\mathbf{x}(t+1) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \quad t = 0,1,2,... \tag{3a}$$

with $$\mathbf{A} \in \Re_+^{n\times n},\ \mathbf{B} \in \Re_+^{n\times m} \text{ and } \mathbf{u}(t) \in \Re_+^{m} \tag{3b}$$

where $\mathbf{x}(t) \in \Re_+^{n}$ is the state of the system at time t, $\mathbf{u}(t)$ is the control and $\Re_+^{n+s}$ is the space of all $n \times s$ non-negative matrices.

Note that **A** and **B** being non-negative matrices represent the necessary and sufficient conditions for a discrete-time linear system with non-negative control to have a non-negative state trajectory for any non-negative initial state.

The class of positive systems (3) considered in this paper is restricted by the following assumptions:

(a) the open-loop transition matrix **A** is monomially similar to the companion matrix,

$$\mathbf{A}_0 = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 \\ \alpha_o & \alpha_1 & \alpha_2 & \cdots & \alpha_{n-2} & \alpha_{n-1} \end{bmatrix} \geq 0 \tag{4}$$

so that $\mathbf{A}_0 = \mathbf{MAM}^{-1} = \mathbf{DPA\ P'\ D}^{-1} = \mathbf{DC\ D}^{-1}$, where $\mathbf{M} \geq 0$ is a monomial matrix, $\mathbf{D} \geq 0$ is a non-singular diagonal matrix, **P** is a permutation matrix and **A** is congruent to the matrix $\mathbf{C} \geq 0$. Since pre- and post-multiplication by a non-singular diagonal matrix 'do not' change the zero, non-zero pattern of a matrix **C** has the same non-zero pattern as $\mathbf{A}_0$ and

(b) the control matrix **B** is an m-monomial matrix.

This well structured mathematical model is applicable in many fields such as, for example, dynamic population model with immigration and other cohort models with applications in industrial engineering (Luenberger[8], Rouhani and Tse[12]). The robot population dynamics model represented by (1) and (2) may be expressed in the matrix-vector form depicted in (3), with

$$\mathbf{A} = \begin{bmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \cdots & \alpha_{n-1} \\ \beta_0 & 0 & 0 & \cdots & 0 \\ 0 & \beta_1 & 0 & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots\beta_{n-2} & 0 \end{bmatrix} \geq 0 \tag{5}$$

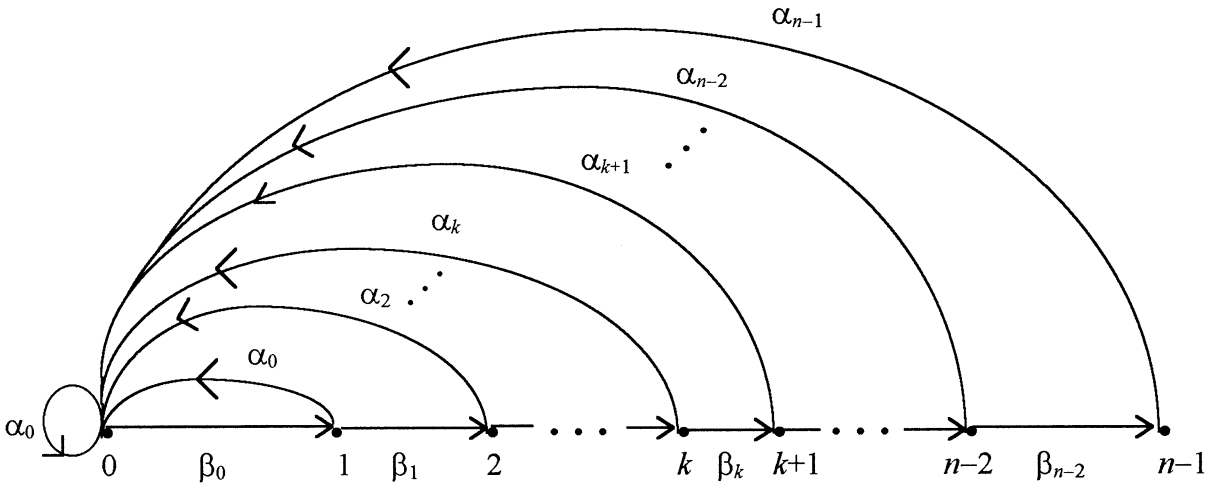


Fig 2: The cohort matrix digraph

which is clearly monomially similar to $\mathbf{A}_0$ in (4), $\mathbf{B} = \mathbf{I}$ the unit $n \times n$ matrix and is therefore a non-negative n-monomial matrix, $\mathbf{x}(t) = (x_0(t)\ x_1(t)\ \ldots\ x_{n-1}(t)) \geq 0$ and $\mathbf{u}(t) = (u_0(t)\ u_1(t)\ \ldots\ u_{n-1}(t)) \geq 0$.

4. Reachability and controllability

The following definitions of (positive) reachability, null-controllability and controllability for positive linear systems can be found in Rumchev and James[9].

The system (3a)-(3b) (and the non-negative pair $(\mathbf{A},\mathbf{B})\geq 0$) is said to be

(a) *reachable* (or controllable-from-the origin) if for any state $\mathbf{x} \in \Re_+^n$, $\mathbf{x} \neq 0$, and some finite t there exists a non-negative control sequence $\{u(s),\ s = 0,1,,\ \ldots\ \ldots\ ,\ t-1\}$ that transfers the system from the origin into the state $\mathbf{x} = \mathbf{x}(t)$;

(b) *null-controllable* (or controllable-to-the origin) if for any state $\mathbf{x} \in \Re_+^n$ and some finite t there exists a non-negative control sequence $\{u(s),\ s = 0,1,,\ \ldots\ \ldots\ ,\ t-1\}$ that transfers the system from the state $\mathbf{x} = \mathbf{x}(0)$ into the origin;

(c) *controllable* if for any non-negative pair $\{\mathbf{x}_o,\ \mathbf{x}\}$ $\mathbf{x} \in \Re_+^n$ and some finite t there exists a non-negative control sequence $\{u(s),\ s = 0,1,,\ \ldots\ \ldots\ ,\ t-1\}$ that transfers the system from the state $\mathbf{x}_o = \mathbf{x}(0)$ into the state $\mathbf{x} = \mathbf{x}(t)$.

The discrete-time positive linear system is controllable if and only if it is reachable and null-controllable (see, eg Rumchev and James[9]) so that controllability implies both reachability and null-controllability and, vice versa, reachability and null-controllability *together* imply controllability. Reachability, null-controllability and controllability are general properties of the system (but not of its environment). They express the *ability* of the system to move in space, this being the non-negative orthant for the class of positive systems under consideration.

Let $D(\mathbf{A})$ be the digraph of an $n \times n$ non-negative matrix $\mathbf{A}$, and constructed as follows. The set of vertices of $D(\mathbf{A})$ is denoted as $N = \{0,1,2,\ldots,n-1\}$ and there is an arc (i, j) in $D(\mathbf{A})$ if and only if $a_{ji} > 0$; the set of all arcs is denoted by U. Notice that by defining $D(\mathbf{A})$ in this way the ith vertex in $D(\mathbf{A})$ corresponds to the $(i+1)$th column in $\mathbf{A}$, $i = 0,1,2,\ldots,n-1$. The digraph of the cohort matrix $\mathbf{A}$ given by (5) is presented in Fig 2.

A *walk* in $D(\mathbf{A})$ is an alternating sequence of vertices and arcs. The walk is called *closed* if the initial and final vertices coincide and *spanning* if it passes through all the vertices of $D(\mathbf{A})$. It is said to be a *path* if all of its vertices are distinct, and a *cycle* if it is a closed path. The path length is defined to be equal to the number of arcs it contains. The number of arcs away from a vertex i is called *outdegree* of i and is written $od(i)$, whilst the number of arcs directed toward a vertex i is called *indegree* of i and is written $id(i)$. Notice that zero columns, respectively zero rows, in $\mathbf{A}$ correspond to vertices j with $od(j) = 0$, respectively to vertices i with $id(i) = 0$, in $D(\mathbf{A})$. Finally, the positive entries in the columns of $\mathbf{B}$ are identified with the corresponding vertices in $D(\mathbf{A})$.

A canonical decomposition of the digraph $D(\mathbf{A})$ into monomial components has been found recently in Caccetta and Rumchev[10]. Monomial components are defined as components of $D(\mathbf{A})$ with outdegrees of each of their vertices equal to at most one. The following monomial components of $D(\mathbf{A})$ are identified in this reference: simple monomial paths (smp), blossoms, monomial trees and bunches. Monomial columns that appear in the reachability matrix $\Re_t = [\mathbf{B}\ \mathbf{AB}\ \mathbf{A}^2\mathbf{B}\ \ldots\ \mathbf{A}^{t-1}\mathbf{B}]$ are due to the monomial components in $D(\mathbf{A})$.

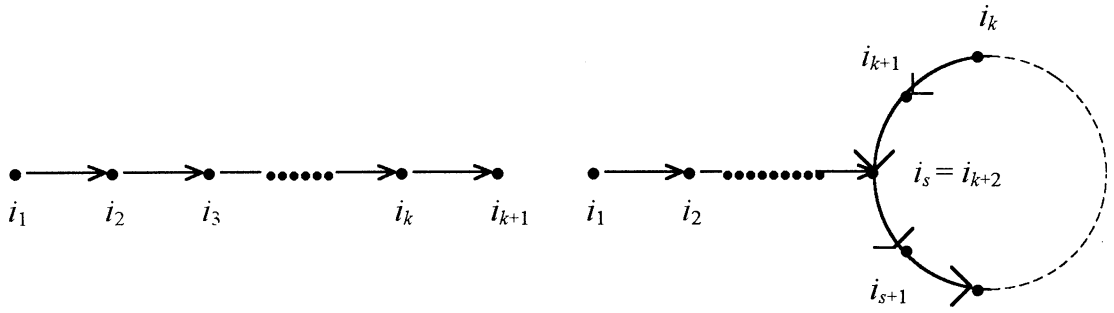


Fig 3: Canonical monomial components: (a) simple monomial path, and (b) blossom

The idea of decomposing the digraph $D(\mathbf{A})$ into monomial components (Caccetta and Rumchev[10]) is quite simple. If all of the outward arcs from vertices with $od(i) \geq 2$ in $D(\mathbf{A}) = (N, U)$ are removed from $D(\mathbf{A})$ then the *reduced digraph* $D^{(o)}(\mathbf{A}) = (N, U^{(o)})$ becomes a *union of disjoint monomial structures* (smp, blossoms, monomial trees and bunches) since $od(i) < 2$ for any vertex $i \in D^{(0)}(\mathbf{A})$. It can be shown that all monomial structures that are in $D^{(o)}(\mathbf{A})$ are also in $D(\mathbf{A})$, and vice versa. Procedures for decomposing monomial trees into smp and bunches into a blossom and simple monomial paths are also developed in Caccetta and Rumchev[10]. Simple monomial paths (Fig 3a) and blossoms (Fig 3b) are the *simplest* (*canonical*) *monomial components* of $D(\mathbf{A})$ and of $D^{(o)}(\mathbf{A})$. It is easy to see from Fig 3a that the indegree of the initial vertex (called origin) of a smp is $id(i_1) = 0$ and the outdegree of the final vertex is $od(i_{k+1}) = 0$. Any part of a simple monomial path with an indegree of the initial vertex or an outdegree of the final vertex equal to one is called a monomial path.

Reachability, null-controllability and controllability criteria in digraph form are given in the following propositions:

Proposition 1 (*Reachability criterion in digraph form*) (Caccetta and Rumchev[10])

Let $\mathbf{A} \geq 0$, and let the associated digraph $D(\mathbf{A})$ have no vertices with $od(i) = 0$. Let also $I_1 = \{i_1^{(1)}, i_1^{(2)}, \ldots, i_1^{(\mu)}\}$ and $J_1 = \{j_1^{(1)}, j_1^{(2)}, \ldots j_1^{(\sigma)}\}$ be, respectively, the sets of all origins (of simple monomial paths and blossoms) and any set of vertices such that $j_1^{(k)} \in C_k$, $k = 1,2,\ldots, \sigma$, where C_k are disjoined cycles in the reduced digraph $D^{(o)}(\mathbf{A})$. Then the pair $(\mathbf{A}, \mathbf{B}) \geq 0$ (and the positive linear system (3a)-(3b)) is reachable if and only if matrix $\mathbf{B}$ contains the submatrix

$$\mathbf{B}^{(0)} = \left[\mathbf{e}_{i_1^{(s)}} \quad , \quad s = 1,2,\ldots,\mu \quad ; \quad \mathbf{e}_{j_1^{(k)}} \quad , \quad k = 1,2,\ldots,\sigma \right].$$

Proposition 2 (*Null-controllability criteria*)
The following are equivalent:

(i) positive linear system (3a)-(3b) (and the pair $(\mathbf{A}, \mathbf{B}) \geq 0$) is null-controllable;

(ii) $\mathbf{A}$ is a nil-potent matrix;

(iii) there are no cycles in the reduced digraph $D(\mathbf{A})$.

The proof of (i) – (ii) can be found in Rumchev and James[9]. It is not difficult to prove (ii) – (iii) using the properties of nil-potent non-negative matrices.

Proposition 3 (Controllability criterion, see Rumchev and James[9])

The positive linear system (3a)-(3b) (and the pair $(\mathbf{A}, \mathbf{B} \geq 0)$ is controllable if and only if it is reachable and null-controllable.

These criteria are used in the next section to study reachability and controllability properties of the cohort model.

5. Controlled evolution

In this section some structures of robot populations described by the cohort model given by (1) and (2) with respect to their reachability and controllability properties are studied.

Case 1: **Totally sterilised robot population**

Production (birth) rate of each generation (cohort) of robots is zero, ie $\alpha_i = 0$ for $i = 0,1,2,\ldots\ldots, n-1$; inflows from other robotic communities to each generation are permitted, ie $B \equiv I$.

The structure of this robot population is given by its digraph $D(\mathbf{A})$ of Fig 4. The vertices i represent the different generations; $od(i) \leq 1$ and hence the reduced digraph $D^{(o)}(\mathbf{A}) = D(\mathbf{A})$. It is seen from Fig 4 that $D(\mathbf{A})$ is a simple monomial path with vertex 1 as origin. Positive entries of the columns of $\mathbf{B}$ are identified with the corresponding vertices. It can be concluded applying Proposition 1, that this structure is reachable since $\mathbf{B}$ contains $\mathbf{e}_1$. As a matter of fact the sterilised robot population does not need migration into generations $i = 1,2,\ldots, n-2$ to achieve any (non-negative) distribution of robots among the different generations from a zero population. The sterilised robot population is also null-controllable according to Proposition 2 since $D(\mathbf{A})$ does

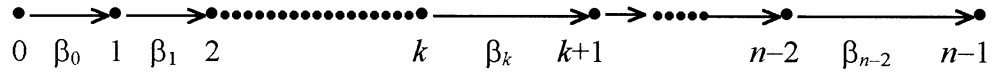


Fig 4: Totally sterilised robot population structure

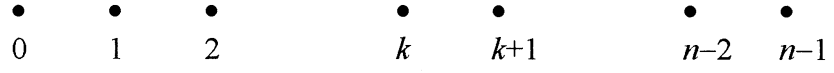


Fig 5: Totally fertile robot population structure

not contain cycles (the cohort matrix **A** is nil-potent with index of nil-potency n). Consequently, the sterilised robot population will become extinct in $n-1$ periods if there is no migration flow.

The sterilised robot population is reachable and null-controllable so according to Proposition 3 it is controllable. A remarkable feature of this population is that we can exercise *complete control over its evolution by building new-generation-robots only*. This conclusion holds true under the assumption that only robots of cohort $(n-1)$ become obsolete, that is $\beta_i > 0$ for $i = 0,1,\ldots, n-1$. Consider now the case when robots, say, of the kth generation G_k become obsolete, ie $\beta_k = 0$ for some k, $0 < k < n-1$, during the whole life-span of the robot community. Then to exercise complete control over the population we need migration from other robot communities into G_{k+1} in addition to building new-generation-robots.

Case 2: **Totally fertile robot population**

Production rates: $\alpha_i > 0$ for $i = 0,1,2, \ldots \ldots, n-1$
Survival rates: $\beta_i > 0$ for $i = 0,1,2, \ldots \ldots, n-2$

The structure of this robot population is represented by its digraph $D(\mathbf{A})$ of Fig 2. It can be seen from this figure that all vertices, $i = 0,1,2, \ldots \ldots, n-1$, have $od(i) = 2$.

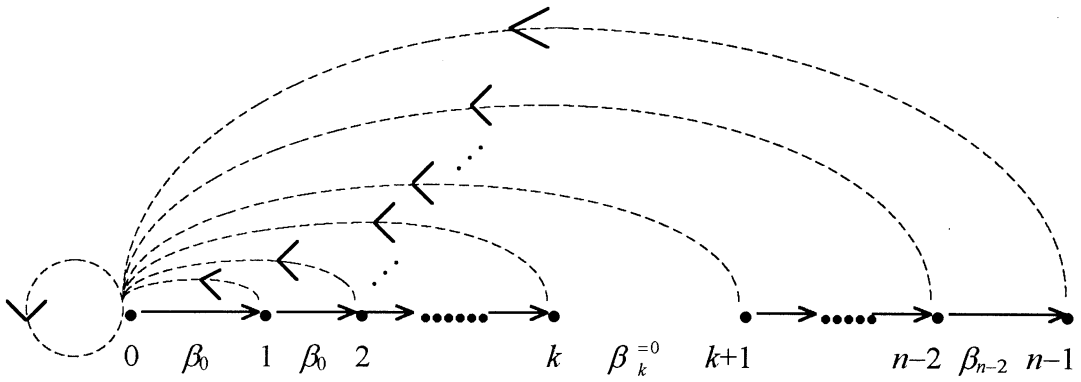
The reduced digraph $D^{(o)}(\mathbf{A})$ (the outward arcs from vertices with $od(i) \geq 2$ removed) is shown in Fig 5. It can be seen that the reduced digraph $D^{(o)}(\mathbf{A})$ is a union of disjoint vertices. Positive entries of the columns of $\mathbf{B} \equiv \mathbf{I}$ are identified with the corresponding vertices. The conditions of Proposition 2 are satisfied and, hence a totally fertile robot population is reachable. This means that totally fertile robot population *needs migration flows* to all generation to achieve any desired (non-negative) distribution of robots among the different generations from a zero population. Since all production rates $\alpha_i > 0$ (the population is totally fertile) the digraph $D(\mathbf{A})$ contains cycles so that **A** is not a nil-potent matrix. Therefore the totally fertile robot population is not null-controllable; that is, it can not be brought to extinction even when there is no migration from other robotic communities. Thus, the *totally fertile robot population* is reachable but not null-controllable and it readily follows from Proposition 3 that it *is not controllable*. This fact is quite understandable, since the production mechanism intrinsic for this robot population is such that it is not possible, by regulating the migration of robots, to achieve any desired (non-negative) distribution of robots among the different generations of robots from any other (initial non-negative) distribution.

Case 3: **Fertile robot population with a generation becoming obsolete during robot life span**

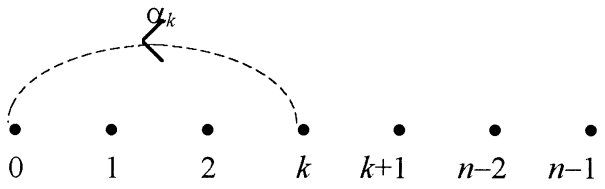
Production rates: $\alpha_i \geq 0$ for $i = 0,1,2, \ldots \ldots, n-1$.
Survival rates: $\beta_k = 0$, $\beta_i > 0$ for $i = 0,1,2, , k-1$, $k+1 ,.., n-2$.

Such a robot population structure is represented by the digraph $D(\mathbf{A})$ shown on Fig 6a, in which the dashed arcs correspond to the production rates α_i. The reduced digraph $D^{(o)}(\mathbf{A})$ (after removing the outward arcs from vertices with $od(i) \geq 2$) is shown in Fig 6b. It can be seen that $D^{(o)}(\mathbf{A})$ is a union of disjoint vertices and a simple monomial path $\{k, (k, 0), 0\}$. The matrix $\mathbf{B} \equiv \mathbf{I}$ contains all the columns required for reachability (see Proposition 1) and therefore this robot population is reachable. Examining the digraph $D(\mathbf{A})$ it readily follows from Proposition 2 that the robot population is not null-controllable, and therefore not controllable (see Proposition 3).

Assume now that the first k generations of robots need some time to mature and so do not contribute to the development of new-generation-robots G_0, that is $\alpha_i = 0$ for $i = 0,1,2, \ldots \ldots, k$. The associated digraph $D(\mathbf{A}_m)$ of this structure is given in Fig 7a and it is not difficult to see, examining the digraph $D(\mathbf{A}_m)$, that the corresponding matrix $\mathbf{A}_m$ is nil-potent, and hence the robot population becomes null-controllable. The reduced digraph $D^{(o)}(\mathbf{A}_m)$ is given in Fig 7b. It contains a smp $\{0, (0, 1), 1, \ldots, (k-1, k), k\}$ and the set of disjoint

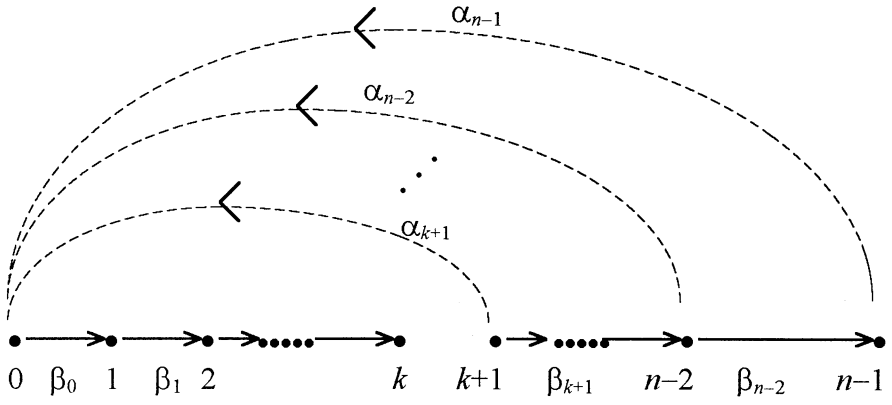


(a)

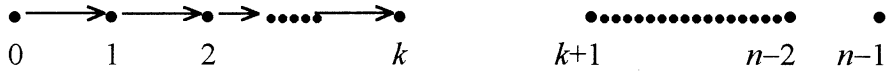


(b)

Fig 6: Fertile robot population structure with a cohort becoming obsolete during robot life span



(a)



(b)

Fig 7: Fertile robot population structure with a cohort becoming obsolete during robot life span and a period needed for maturity

vertices $\{ k+1, \ldots, n-1 \}$. According to Proposition 1 the matrix $\mathbf{B}^{(o)} = (\mathbf{e}_1, \mathbf{e}_{k+1}, \ldots, \mathbf{e}_{n-1})$ makes the pair $(\mathbf{A}_m, \mathbf{B}^{(o)})$ reachable. But $\mathbf{B}^{(o)}$ is a submatrix of $\mathbf{B} \equiv \mathbf{I}$. So the *fertile robot population with a cohort becoming obsolete during the robot life span and a period needed for maturity is controllable.*

6. Conclusions

A new dynamic model of robot populations is developed. The model is specified for different types of robot population structures: totally sterilised robot population, totally fertile robot population and partially fertile/sterilised robot populations. Reachability and controllability properties of such populations of robots are studied and analysed in detail. These fundamental properties have an important impact on their evolution with a number of direct consequences for planning the future societies of intelligent autonomous robots. The results obtained increase our understanding of how to control the evolution of such robot societies.

References

(1) Sekiyama K and Fukuda T (1996) Self-organizing control strategy for group robotics, *Advanced Robotics*, **10**, 6, pp 637-658.

(2) Takadama K, Hajiri K, Nomura T, Nakasuka S and Shinohara K (1998) Organizational knowledge on formation in multiple robots learning, *In: Sugisaka M (ed) Proc Third Int Symp Artif Life and Robotics*, Beppu, Japan, **2**, pp 397-401.

(3) Agah A and Bekey GA (1994) Autonomous mobile robot teams, *In Proc AIAA/NASA Conf on Intelligent Robots in Field, Factory, Service and Space*, Houston, TX, **1**, pp 246-251.

(4) Yoshida E, Arai T and Ota J (1998) Local communication of multiple mobile robots: design of group behaviour for efficient communication, *Advanced Robotics*, **11**, 8, pp 759-779.

(5) Suzuki T, Fujii T, Asama H, Yokota K, Kaetsu H and Endo I (1998) A multi-robot teleoperation system using the Internet, *Advanced Robotics*, **II**, 8, pp 781-797.

(6) Hirosi S (1992) A dispute over robots. Robots of the future, *Advanced Robotics*, **6**, 2, pp 231-241.

(7) Agah A (1996) Robot teams, human workgroups and animal sociobiology: a review of research on natural and artificial multi-agent autonomous systems, *Advanced Robotics*, **10**, 6, pp 523-545.

(8) Luenberger D (1979) Introduction to dynamical systems: theory, models and applications, *Wiley, New York.*

(9) Rumchev V and James DJG (1989) Controllability of positive linear systems, *International Journal of Control*, **50**, 3, pp 845-857.

(10) Caccetta L and Rumchev VG (1998) Reachable discrete-time systems with minimal dimension control sets, *Dynamics of Continuous, Discrete and Impulsive Systems* (to appear)

(11) Szidarovsky F and Bahill AT (1992) Linear systems theory, *CRC Press, London.*

(12) Rouhani R and Tse E (1981) Structural design for classes of positive linear systems, *IEEE Transactions on Systems, Man and Cybernetics*, **11**, pp 126-134.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Ola: What Goes Up, Must Fall Down

Henrik Hautop Lund Jens Aage Arendt Jakob Fredslund Luigi Pagliarini

LEGO Lab
InterMedia, Department of Computer Science
University of Aarhus, Aabogade 34, 8200 Aarhus N., Denmark
http://www.daimi.aau.dk/~hhl
hhl@daimi.aau.dk

Abstract

We have made a robot soccer model using LEGO Mindstorms robots, which was shown at RoboCup98 during the World Cup in soccer in France 1998. For the robot soccer model, we constructed a stadium out of LEGO pieces, including stadium light, rolling commercials, moving cameras projecting images to big screens, scoreboard and approximately 1500 small LEGO spectators who made the "Mexican wave" as known from soccer stadiums. These devices were controlled using the LEGO Dacta Control Lab system and the LEGO CodePilot system that allow programming motor reactions which can be based on sensor inputs. The wave of the LEGO spectators was made using the principle of *emergent behaviour*. There was no central control of the wave, but it emerges from the interaction between small units of spectators with a local feedback control.

Introduction

World Cup and regional games (such as European Championship) in soccer always attract very big crowds of passionate fans. The passionate fans on the grandstands are part of what makes soccer an animated game. For instance, the fans are known to make big choreographic scenery such as the one on figure 1. The impressive image arises when each spectator holds up a coloured piece of paper that he/she has been handed out from an organising group of fans. We can interpret this organising group of fans as a central control that decides what colour should be handed out to the individual spectator. Without this central control, the choreographic scenery would fail. The "Mexican wave" that is made when spectators stand up and sit down does not have such a central control. The wave is initialised when a couple of spectators anywhere on the stadium decide to make the stand up + sit down movement, and some nearby spectators go with the others. There is no central control to tell the individual spectator to do a specific thing at a given time, rather it is an emergent behaviour.



Figure 1. Impressive choreographic scenery made by Lazio's tifosi in Curva Nord of the Olympic Stadium in Rome. The scenery is constructed when each spectator holds up a coloured piece of paper.

Emergent behaviour is an interesting phenomenon that can be observed in natural systems. We define emergent behaviour as being the behaviour of a system that is the product of interaction between smaller sub-systems. The emergent behaviour is of higher complexity than the sum of the behaviours of the smaller sub-systems. The reason that behaviour of higher complexity than the sum can emerge is the interaction between the sub-systems.

Emergent behaviour is known from flocks of birds, schools of fish and herds of land animals. When observing a flock of birds, we will notice that there is no apparent leader in the flock and there appears to be no central control of motion. The motion of the flock might seem complex and at times random, but on the other hand, it also appears synchronous. The motion of a flock of birds is an example of emergent behaviour and can be modelled as such. Reynolds [Reynolds 1987] has made an impressive study of the general motion of flocks, herds, and schools in a distributed behavioural model with the goal of using this to model flocking in computer graphics. Recently, similar models have been used in the Disney movie *Lion King* for a wild-beast stampede and to produce photo-realistic imagery of bat swarms in the feature motion pictures *Batman Returns* and *Cliffhanger*. Reynolds calls his simulated bird-like organisms *boids* (bird-oids). The boids are controlled by three primary rules:

1. Collision Avoidance: avoid collision with nearby boids
2. Velocity Matching: attempt to match velocity with nearby boids
3. Flock Centering: attempt to stay close to nearby boids

The three rules are local in the sense that a boid only has knowledge about nearby boids and there is no global knowledge like size or centre of the flock. For instance, Flock Centering is achieved by having boids to perceive the centroid of nearby boids only. This actually gives the advantage of allowing for bifurcation: the flock can split around an obstacle in the moving direction, since the boids only tend to stay close to nearby flock-mates.

In general, the phenomenon of emergent behaviour is fundamental in a number of artificial life systems. Artificial life tries to synthesise life with a bottom-up approach by using small building blocks that emerge to a complex system by their interaction. For instance, emergence is used as the basic principle in cellular automata, and in a sense, emergence is also one of the basic principles behind the success of artificial neural networks. Artificial neural networks are built from units (neurons) and connections between units. Each unit has a simple processing capability and the connections have propagating abilities. But the interaction between many units with simple processing capability results in a more complex behaviour than just the sum of the processing capabilities. In fact, this was used to refuse the criticism of neural networks put forward by Minsky and Papert [Minsky and Papert, 1968] when Rumelhart, Hinton, and Williams showed that neural networks can indeed solve the XOR problem [Rumelhart et al., 1986].

Emergent Behaviour in Reality

When using emergent behaviour in real world models, there are a number of pitfalls that we have to be aware of. When we look at Reynolds' boid model, we notice that only local knowledge of neighbours is used, so the model might appear appropriate for control tasks for autonomous agents in the real world. However, it is not clear how to obtain even the local knowledge that is available for the simulated boids. For instance, we have to solve the question of how to measure *nearby* (distance and direction). This demands an advanced sensor that can measure distance and direction, and at the same time identify an object as being a neighbour (and, for instance, not an obstacle). The task is worsened further by the demand for doing this in real time with moving objects.

There are other significant differences between a simulation model and a real world implementation that we have to take into account. For instance, the actuators will produce friction and there will be a whole range of noise issues that makes it very difficult to transfer an idealised model from simulation to the real world. In some cases, it will be possible to transfer models from simulation to reality [Miglino et al., 1995; Lund and

Miglino, 1996; Jakobi, 1998]. This is done by very careful building of a simulator that models the important characteristics of the real device and the way that real world noise interferes with this device.

In our emergent behaviour model, we work directly in the real world, so we avoid the problems of difficulties in transfer from an idealised model to the real world. Our model therefore has to work with the noise, friction, etc. that exists in the real world.

RoboCup and LEGO Robot Soccer

In the summer of 1998, we had to go to Paris during the World Cup in soccer to demonstrate LEGO Mindstorms robots playing soccer at RoboCup'98. RoboCup is an international initiative to promote artificial intelligence robotics and the task of robot soccer as a landmark project [Kitano et al., 1997]. As a landmark project, RoboCup differs from earlier artificial intelligence landmark problems, such as constructing an artificial chess player. One of the main differences is that robot soccer players have to play in the real world, where the chess play can be viewed as an idealised world, in which there is no need to address the problems of perception and noise in the real world. Essentially, the differences are similar to the differences between a simulated model of emergence and a real world model. In general, the differences can be summarised as shown in Table 1 (reprinted with permission from H. Kitano).

Table 1. Differences between the classical artificial intelligence landmark project of constructing an artificial chess player and the landmark project of constructing a team of robot soccer players [Kitano et al., 1997].

	Chess	Robot soccer
Environment	Static	Dynamic
State change	Turn taking	Real time
Information accessibility	Complete	Incomplete
Sensor readings	Symbolic	Non-symbolic
Control	Central	Distributed

We constructed team of LEGO Mindstorms robot soccer players to play a demonstration tournament during RoboCup'98. The LEGO Mindstorms robot soccer players are described elsewhere, so what follows will only be a short description of the physical set-up.

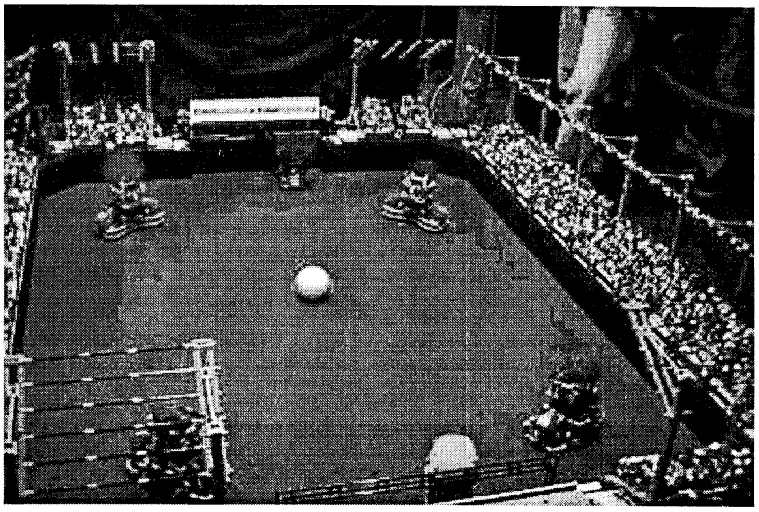


Figure 2. The LEGO robot soccer set-up. There is one goalkeeper and two field players on each team (one red and one blue team). The stadium has light towers, scanning cameras that project images to large monitors, scoreboard, rolling commercials, and almost 1500 small LEGO spectators that make the "Mexican wave". © H. H. Lund, 1998.

Each team consisted of one goalkeeper and two field players. The goalkeeper was controlled with a LEGO CodePilot, while the two field players were constructed around the LEGO Mindstorms RCX (Robot Control System). Each player had two independent motors to control two wheels to make the robot move around on the field, and one motor to control movement of the robot's mouth (so that it could "sing" the national anthem and "shout" when scoring a goal). A player had three angle sensors to detect the motion of wheels and mouth. All parts of the robots except for batteries and coloured hair to indicate the team were original LEGO elements (LEGO Dacta, LEGO Mindstorms, LEGO Technic).

In order to put the robot soccer play into a stimulating context, we built a whole LEGO stadium (see Figure 2). The stadium had light towers (with light) in each corner, and these towers also hold infra-red transmitters that could transmit information from a host computer to the RCXs. In one end, there was a scoreboard that could be updated when a goal was scored via an interface with the LEGO Dacta Control Lab (see Figure 3). Over each goal, there was a rolling commercial sign that held three commercials that were shown in approximately 30 seconds each before the sign would turn to the next commercial. The control of the two rolling commercial signs was made with the LEGO CodePilot. A camera-tower with a small b/w camera was placed in one corner. The camera (controlled from a CodePilot) could scan to the left and the right of the field, while displaying the image to the audience on a large monitor. Another camera was placed over the sideline on one side and should scan back and forth following the ball (see Figure 3). Also this camera image was displayed on a large monitors, and its control was made from LEGO Dacta Control Lab.

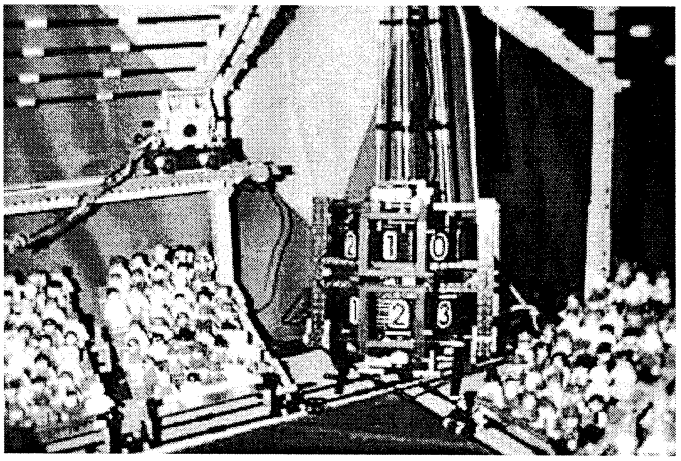


Figure 3. The scoreboard and one of the small cameras. The camera runs up along the sideline, while projecting the images to a large monitor. © H. H. Lund, 1998.

The LEGO spectator wave

Apart from robot soccer players, cameras, rolling commercials, and scoreboard, we had placed almost 1500 small LEGO spectators on the grandstands. Our idea was to have all these spectators make the "Mexican wave" as we see soccer fans make at real World Cup matches. Our first intuition was to make a central control with one block running underneath the spectators and pushing the single spectator up when reaching him/her. Then, when the block had passed, the spectator would move down again. As the block moved around on the grandstands, it should therefore produce a wave when pushing spectators up and allowing them to move down again.

We made a prototype of this first model with a block running underneath the spectators. The prototype produced a wave of the spectators, but it had some deficiencies. First of all, the central timing control made the wave stop if the block became stuck for some time, and, secondly, the big LEGO spectators that we

were using would easily get wrapped up together. The second problem was a pure technical problem that we could soon solve using smaller LEGO figures and mounting them into sections. By doing this, the position of the LEGO figures would be fixed and they would not sway from one side to the other like the larger LEGO figures tended to do when they were moving up and down.

However, the first problem was of a more fundamental nature, and we would have to change the control perspective. The control approach was directly opposing what had been lectured about in the lectures, but this was apparently not realised until observing the implemented prototype. The control of the block was a classical example of timing control, or *open-loop control*, in which the reaction of the agent is dependent on an internal timing. The block would move forward for a pre-defined time and then move backward for the same pre-defined time. If something unexpected happened during the movement, the block would not be able to respond to this change of circumstances, since its actions depended on a timer rather than on environmental circumstances. The approach had been to idealise the world and assume a smooth movement of the block (essentially believing in a simulated model of the real world). On the other hand, the lectures had taught the students about *feedback control*, in which the behaviour of an agent is dependent on the feedback from the environment, so that the agent can react on changed environmental circumstances rather than having a fixed behaviour that depends on a timing.

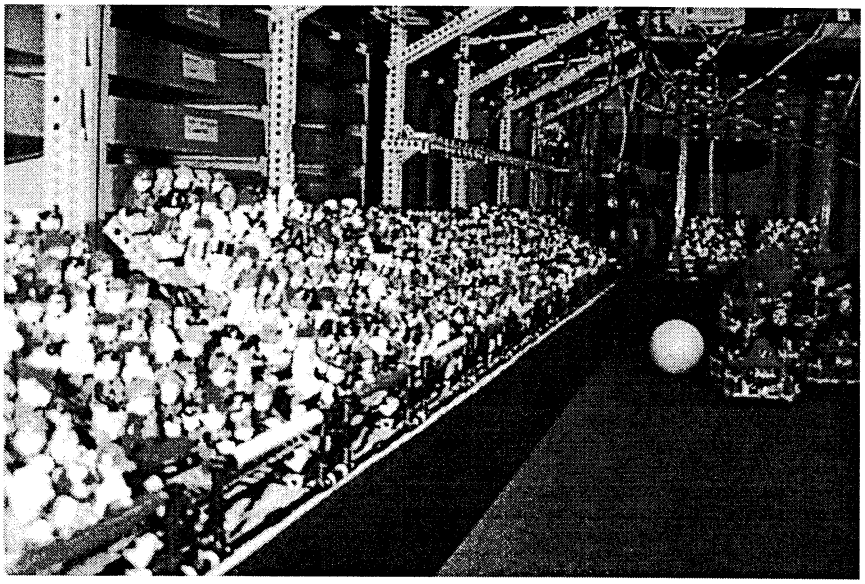


Figure 4. The LEGO spectator wave. When the switch sensor is released, the next section of LEGO spectators will start to move upwards. © H. H. Lund, 1998.

The control of the wave was now changed to a feedback control, and the idea was to allow the wave to *emerge* from the interaction between the different sectors of spectators that each had their own, local feedback control. In this case, the implementation of a system with emergent behaviour should be possible in the real world, since there would be no demand of advanced sensing. In fact, a switch sensor for each section of LEGO spectators turned out to be enough (see Figure 5). The idea was that the movement of one section should be dependent on sensing what the adjacent section was doing. If a section was moving upwards, then the section to the right should sense this and start to move upwards itself. The section would fall down when reaching the top position (in this way, we used the principle that *what goes up, must fall down*). This was built by placing a switch sensor under each section and connecting this switch sensor to the control unit (a LEGO CodePilot) of the next section. In resting mode, the switch sensor would be pressed by the section of spectators above it, but when this section started to move upwards, the switch sensor would no longer be pressed. This triggered the simple control program in the next section to start moving the section of LEGO spectators upwards. In pseudo-code, the control program of each section could be as follows:

```
Section N control:
      if (Switch(N-1)=false) then turn on motor
      else turn off motor
```

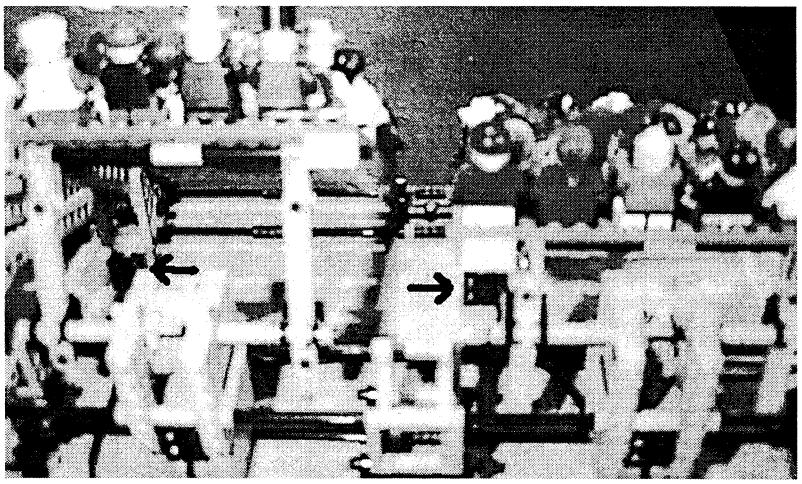


Figure 5. The movement of a section is triggered by the switch sensor mounted underneath the adjacent section. The two arrows show the placement of the switch sensors. The left section of spectators has risen, so the switch sensor is no longer pressed. The control of the right section will notice this, and start to move upwards (immediately after this photo was taken). © H. H. Lund, 1998.

This very simple control allows the wave to *emerge* when one section is triggered from the external to move upwards. In the actual implementation in the LEGO CodePilot language, it was however necessary to use a timer, since the time slice of the CodePilot is so small, that the above pseudo-code program would result in the section barely moving upwards before the motor would be turned off. So when it was sensed that the switch was no longer pressed, the control would turn the motor one direction for 0.5 seconds and then the other direction for 0.5 seconds. It would have been more sensible to have a switch sensor on the top position that the section should reach, and then base the time of upward and downward movement on the feedback from the top and the bottom sensors. But since the LEGO CodePilot has only one input channel (see Figure 6), we opted for the timing solution. However, it must be noted that this timing is very different from the timing in the first prototype, since here, even though the timing of a section might have been wrong, the section would still lift itself for some time and therefore the next section would be triggered. In a sense, we are setting the time-slice to 0.5 seconds and use the pseudo-code program.

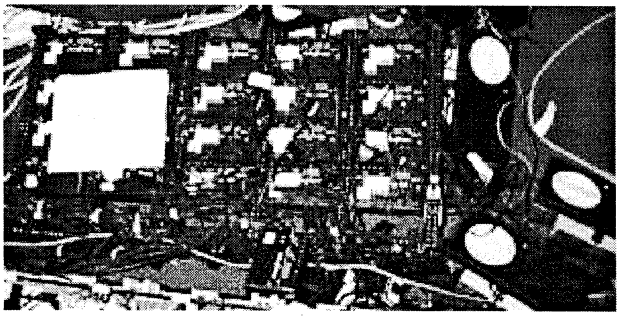


Figure 6. The mixer. 16 LEGO CodePilots were used to construct the dynamics of the "Mexican wave". Each CodePilot had one switch sensor and one motor connected. © H. H. Lund, 1998.

The feedback control was used and the wave emerged from the interaction between the simple units. It was run numerous times daily in Paris for a week without any need for refinements apart from a couple of changes of physical aspects (once or twice, a section got stuck).

Conclusion

We used emergent behaviour to construct a "Mexican wave" of LEGO spectators for the LEGO robot soccer demonstration. The wave emerged from the interaction between sections of LEGO spectators, each with its own simple control. The control was based on feedback from the local environment. Since sensing was straightforward with one switch sensor for each section, it was fairly easy to implement a real world emergent behaviour. Under other circumstances, it might be more difficult, since more advanced sensing might be necessary, and we cannot guarantee that the desired real world behaviour will emerge. Therefore, the study of real world emergent behaviour is important in order to identify the circumstances that will lead to successful results.

Acknowledgements

Students from the LEGO Lab made a highly valuable work in the LEGO robot soccer RoboCup'98 project. Lars Nielsen from LEGO Media worked as a very helpful partner who gave numerous valuable ideas. Ole Caprani participated in a number of inspiring discussions. Thanks to H. Kitano for collaboration on the Paris RoboCup'98 event.

References

[Jakobi, 1998] N. Jakobi. The Minimal Simulation Approach to Evolutionary Robotics. In T. Gomi (ed.) *Proceedings of ER'98*, AAI Books, Kanata, 1998.

[Kitano et al., 1997] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, and H. Matsubara. RoboCup. A Challenge Problem for AI, *AI Magazine*, 73-85, spring 1997.

[Lund and Miglino, 1996] H. H. Lund, and O. Miglino. From Simulated to Real Robots. In *Proceedings of IEEE 3rd International Conference on Evolutionary Computation*, IEEE Press, NJ, 1996.

[Miglino et al., 1995] O. Miglino, H. H. Lund, and S. Nolfi. Evolving Mobile Robots in Simulated and Real Environments. Artificial Life 2(4), 417-434, 1995.

[Minsky and Papert, 1968] M. Minsky and S. Papert. *Perceptrons*. MIT Press, Cambridge, MA, 1968.

[Reynolds, 1987] C. W. Reynolds. Flocks, Herds, and Schools: A Distributed Behavioral Model. *Computer Graphics*, 21(4), 25-34, 1987.

[Rumelhart et al., 1986] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning Representations by Back-propagating Errors. *Nature*, 323, 533-536, 1986.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Dynamic Bidirectional Associative Memory Using Chaotic Neurons

Ju-Jang Lee
Department of Electrical Engineering
Korea Advanced Institute of Science and Technology
373-1 Kusong-Dong Yusong-Gu, Taejon 305-701, KOREA
FAX : +82-42-869-3410 Email : jjlee@ee.kaist.ac.kr

Abstract— **A dynamic bidirectional associative memory (DBAM) with chaotic neurons as nodes is proposed in this work. Learning algorithm based on Pontryagin's minimum principle makes the DBAM is equivalent to any other BAM reported so far. The input selection mechanism gives the DBAM additional ability for multiple memory access, which is based on the dynamics of the chaotic neuron.**

I. Introduction

Associative memories are important neural network models which can be employed to model human thinking and machine intelligence by association. An Associative memory can store a set of patterns. When the associative memory is presented with an input pattern, it responds by producing one of the stored patterns that closely resembles or relates to the input pattern. Hence, the recall (or retrieval) of a pattern is through association of the input pattern with the information memorized. Such memories are also called content-addressable memories in contrast to the traditional memories in digital computers.

Associative memories can essentially be classified into autoassociative memories and heteroassociative memories. Autoassociative memories are in general fully interconnected neural networks which can store multiple stable states [1]-[3]. Each neuron is connected to all the other neurons in the network with symmetric connection strengths. Though autoassociative memories are useful tools for various applications, they tend to produce spurious stable states and highly complex connections.

As an extension of autoassociative memories, heteroassociative memories have been developed. Adaptive resonance theory (ART) [4] and the Bidirectional associative memory (BAM) [5][6] are typical examples. In place of unidirectional association, heteroassociative memories in general perform bidirectional association. They associate an input pattern with a different stored output pattern of a stored pattern pair.

Owing to its lowest connection complexity, guaranteed convergence and stronger error-correction capability, BAM has attracted particular attention in neural network research. Extended on the BAM framework, a number of improvements have been made. Some of these models enhance the BAM architecture to improve the performance by the addition dummy neurons [7] [8], more layers [9], or interconnections among neurons within a layer [10], while others use new learning algorithms to improve the performance [11]-[14].

All of these models assume logical symmetry for the interconnections making the weights from the X-layer to the Y-layer the same as those from the Y-layer to the X-layer. Among the symmetrical models, the symmetrical BAM (SBAM) using the Hamming stability learning algorithm (SBAM) provides superior performance [12]. However, the logical symmetry of interconnections not only severely hampers the efficiency of BAM's in pattern storage and recall capability, but also limits their usefulness in knowledge representation and inference systems. To overcome the drawback of symmetrical interconnections, an asymmetrical BAM model (ABAM) has been proposed in [15]. However, the learning algorithm associated with the ABAM model requires linear independence of stored patterns, which limits the storage capacity. So far, there are no BAM's that can store more pattern pairs than the number of neurons in their layers.

A Generalized Bidirectional Associative Memory (GBAM) [16] has recently been reported, which is able to store more pattern pairs than the number of neurons in their layers. The GBAM is superior to all the other bidirectional memories proposed so far. This model is asymmetric and is trained using a set of pattern pairs extended from the pattern pairs to be stored to guarantee asymptotic stability. In general, accurate learning methods like back propagation, with additional elaborate data preprocessing and data preparing, have the advantage of precise recall, high memory capacity, and less spurious memories but have the disadvantages of long learning time and small attractive basins. The coarse learning methods like the correlation learning in Kosko's BAM (KBAM) is simple but suffer from low memory capacity and many spurious memories.

All the associative memories with neurons of Hopfield type are internally static at the neuron-level, so that an input pattern in X-layer cannot be learned with multiple output patterns in Y-layer. Learning process itself is impossible because multiple access with a key pattern cannot be represented by any function. By increasing the number of layers, multiple memory pairs could be stored and retrieved [17]. However, for an n-tuple association it requires n layers and it wastes a lot of space.

In this paper, a dynamic bidirectional associative memory (DBAM) is proposed, in which each neuron is a chaotic neuron. Basically, this DBAM has the same structure as any other kind of BAM described above, except that the neurons have internal dynamics acting synchronously each

other and beside the *interlayer* weights *intralayer* weights are established. The proposed DBAM has the same capability of the GBAM and additionally has a salient feature, the capability for multiple memory access (MMA). Due to the internal dynamics of the chaotic neuron, as the learning progresses it converges to a certain periodic orbit.

II. Chaotic neuron as an element of DBAM

The chaotic neuron [18] is governed by the following equations,

$$p(t+1) = \text{sign}(q(t+1)), \tag{1}$$

$$q(t+1) = kq(t) + \alpha s(q(t), a(t))g(q(t)) + a(t), \tag{2}$$

where $p(t+1)$ is the neuron output, $q(t+1)$ is the internal state of a neuron, k is the damping factor of the refractoriness, α is a signed scaling parameter, $s(q(t), a(t))$ is a switching function, $a(t)$ is the strength of the input at t, and g is a Gaussian function with zero mean. The switching function, $s(q(t), a(t))$, is defined as

$$s(q,a) = \begin{cases} 1 & \text{if } q(t) - q(t-1) > 0 \\ -1 & \text{if } q(t) - q(t-1) < 0 \\ -1 & \text{if } q(t) - q(t-1) = 0 \text{ and } a(t) \geq 0 \\ 1 & \text{if } q(t) - q(t-1) = 0 \text{ and } a(t) < 0. \end{cases} \tag{3}$$

As seen from Eq. (2) the internal state at $t+1$, $q(t+1)$, depends on the input, the linearly scaled current internal state, $q(t)$, and the nonlinear self–feedback. The nonlinear self–feedback changes its sign depending on the variation tendency of the current internal state. Therefore the nonlinear term is positive in the deceasing phase and is negative in the increasing phase. Moreover, the characteristic of $s(q(t), a(t))g(q(t))$ shows hysteresis. In case of neglecting the nonlinear self–feedback term, so that it has little effect on summation with the other terms, the dynamics of the internal state become very simple. When it is dominant, however, the following state undergoes excitatory and inhibitory self–feedback and in turn which leads the internal state to reveal complex dynamics.

The functional role of this neuron is to partition the input space into two regions in bipolar state with an ambiguous intersection. For positive inputs the response tends to converge to 1, whereas for negative inputs the response tends to converge to -1. However, for inputs near to zero there is ambiguity in partitioning.

The characteristics of chaotic neuron is shown in Figure 2. The firing rate of $p(t)$ in Fig. 2(c) represents the frequency of visits made in the positive region of $q(t)$. The firing rate in the biological system is similar to the PWM signal used to control a motor neuron [19]. Its implementation to derive a collective computation model can be found in [20]. The internal state, $q(t)$, is almost the same as the input $a(t)$, when $a(t)$ is far from zero. The dynamics of $p(t)$ undergoes type–I intermittency with respect to $a(t)$. The output of a chaotic neuron has intervals in $a(t)$ in which the firing rates are constant. Constant firing rate means a periodic orbit of $p(t)$ at the corresponding input. Time histories of the output patterns with respect to $a = 10^{-5}$ and $a = -10^{-5}$ are shown in Figure 3. Comparing with Fig. 2(c), it can be observed that a constant firing rate of $1/2$ means a period–6 orbit in which three successive orbits are fired followed by three other orbits not fired when $a = 10^{-5}$. Similarly, constant firing rate $2/3$ means a period–6 orbit in which four successive orbits are fired on followed by the two other orbits fired off when $a = 10^{-3}$. A period–5 orbit shows constant firing rate $4/5$ in which four successive orbits are fired followed by an orbit fired off for $a = 10^{-1}$. Since the ouputs with $a = +\epsilon$ and $a = -\epsilon$ fire mutually exclusively, this feature is utilized to realize double memory access.

III. Interlayer weights learning

For the multiple memory pairs, (x^μ, y^{μ_1}) and (x^μ, y^{μ_2}), bipolar representation of the summed multiple memory is denoted by y^μ, i.e., $y^\mu = \text{sign}(y^{\mu_1} + y^{\mu_2})$. The gradient information between y^{μ_1} and y^{μ_2} is required for input selection, and the difference between the two is denoted as $\Delta y^\mu = (y^{\mu_2} - y^{\mu_1})/2$.

Define the support on a neuron, Y_i, in Y–layer, from the m neurons of the X–layer as $\sum_{j=1}^{m} w_{ij}^1 x_j$ and is denoted by $s(i|x)$. It is the net input in a conventional BAM, which represents the weighted contribution of the firing states of the neurons, X_j. The recalled status, y_i is taken as the sign of $s(i|x)$ with unity magnitude. With the concept of support, the energy associated with the Y–layer is defined as

$$E_Y(w^1; y|x) = -y^T w^1 x = -\sum_{i=1}^{n} y_i s(i|x). \tag{4}$$

Learning the weights can be considered as an optimization to minimize the energy. Thus, Pontryagin's minimum principle plays an important role in the learning phase. The energy relation in the Y–layer is

$$E_Y(w^1; \tilde{y}^\mu|\tilde{x}^\mu) \geq E_Y(w^{1*}; y^\mu|\tilde{x}^\mu) \quad \text{for learning } w^1, \tag{5}$$

where * denotes optimal quantity, $\tilde{x}^\mu \in H_1(x^\mu)$, $\tilde{y}^\mu \in H_1(y^\mu)$, and $H_1(v)$ consists of vectors with Hamming distance from v not to exceed unity. Equality holds, if and only if $\tilde{y} \equiv y^\mu$. This principle for X layer can be obtained similarly. Randomly chosen initial weights are updated until the above energy relations for all neurons and all patterns are satisfied,

$$\Delta w_{ij}^1 = \begin{cases} y_i^\mu \tilde{x}_j^\mu & \text{if } y_i^\mu s(i|\tilde{x}^\mu) \leq 0 \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

Weights in w^2 are updated in the same fashion.

The association pairs for learning w^1 are composed of the pairs in such a way that each pattern in $H_1(x^\mu)$ matches y^μ, while each pattern in $H_1(y^\mu)$ matches y^μ for learning w^2. For this reason, the asymptotic stability is guaranteed in the sense of the stability discussed in [16].

IV. Input selection for multiple association

The information about $(x^\mu, \Delta y^\mu)$ is stored in some neurons, Y_i, in the form of input. For multiple associaton a

stable pair (x^μ, y^μ) is retrieved with a key pattern x^μ, and then the dynamic nature of chaotic neurons makes the y^μ to bifurcate into y^{μ_1} and y^{μ_2}.

Define sets containing the indices for the elements of Δy_i. The elements of Δy_i take the values of 1 and -1 and thier indices are represented as $\mathbf{1}$ and $\bar{\mathbf{1}}$. That is, $\mathbf{1}^{\Delta y^\mu} = \{i \mid \Delta y_i^\mu = 1\}$ and $\bar{\mathbf{1}}^{\Delta y^\mu} = \{i \mid \Delta y_i^\mu = -1\}$. For storing the pairs, (x^μ, y^{μ_1}) and (x^μ, y^{μ_2}), the index sets $i_1 = \mathbf{1}^{\Delta y^\mu}$ and $i_2 = \bar{\mathbf{1}}^{\Delta y^\mu}$ can be determined so that the input selection mechanism yeilds,

$$a_{Yi}(t) = \begin{cases} +\epsilon & \text{if } s(i|x^\mu) = \sum_{j=1}^m w_{ij}^1 x_j(\tau) \text{ and } i = i_1 \\ -\epsilon & \text{if } s(i|x^\mu) = \sum_{j=1}^m w_{ij}^1 x_j(\tau) \text{ and } i = i_2 \\ \sum_{j=1}^m w_{ij}^1 x_j(\tau) & \text{otherwise,} \end{cases} \tag{7}$$

where ϵ is an arbitrarily small positive constant, $t \in \{1, 2, \cdots, T\}$ denotes the discrete time related to the neuron dynamics, and τ is the global discrete time variable in view of DBAM. Input selection for X_j can be determined similarly.

In the retrieving phase, the outputs of X–layer and Y–layer are

$$x_j(\tau + 1) = p_{Xj}(T) \tag{8}$$

and

$$y_i(\tau + 1) = p_{Yi}(T), \tag{9}$$

where T is the last time t of the neuron dynamics.

V. Results and discussion

Simulations are performed for the 26 english alphabet pairs of size 49. Each uppercase letter is associated with its lowercase conterpart. Since the learning process guarantees asymptotic stability, it recalls all the patterns perfectly when the input pattern is not noisy. For input patterns with 8% noise, 14 correct recalls were made (Figure 5), which is superior to, or at least, equivalent to any other result reported so far.

Next we investigated multiple association capability. Table 1 shows the original stored pairs and the learned pairs with asymmetric weights, w^1 and w^2, for three cases. Bipolar representations of the summed multiple memory at the learning phase are denoted by $x^\mu = \text{sign}(x^{\mu_1} + x^{\mu_2})$ and $y^\mu = \text{sign}(y^{\mu_1} + y^{\mu_2})$. Because of the bidirectionality of DBAM, an endurable noisy input pattern stabilizes to take out a proper pattern pair. After aquiring a consistent pair, the bidirectional propagation is terminated and two memories are sequentially retrieved in the Y–layer. The retrieving sequence depends on the magnitude of ϵ, and the pattern y^{μ_1} comes out first. Table 2 shows the recalling sequences of DBAM for three cases. These results are delineated in Figures 6 through 8.

Intralayer weights can make neurons to be competitive. As seen in Figure 8, $w_{12}^y = -1.0$ and $w_{21}^y = -0.6$ confine the outputs to 1 or -1 when $a = \pm\epsilon$. Using these intralayer weights, a double association becomes a single association. For storing the pairs (x^μ, y^{μ_1}) and (x^μ, y^{μ_2}), these intralayer weights restrict the output pattern in Y-layer to y^{μ_1}, while $w_{12}^y = -0.6$ and $w_{21}^y = -1.0$ restricts the output pattern in Y-layer to y^{μ_2}. Thus, the proposed DBAM can be thought of as a model for concentration and distraction in human–thinking. In addition, the on–line input learning of chaotic neurons during the recalling process has a similar effect on concentration and distraction, because an increasing input tends to fire more frequently (see Figure 2(c)).

If the temporal sequence retrieval model similar reported in [21] is utilized with the proposed DBAM, a very interesting spatio–temporal sequences with a key pattern will resulted in.

References

[1] M. A. Cohen and S. Grossberg, "Absolute stability of global pattern formation and parallel memory storage by competitive neural networks," *IEEE Trans. on System, Man and Cybernetics*, Vol. 13, pp. 815–826, 1983.

[2] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Natl. Acad. Sci.*, Vol. 79, pp. 2554–2558, 1982.

[3] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two–state neurons," *Proc. Natl. Acad. Sci.*, Vol. 81, pp. 3088–3092, 1984.

[4] G. Carpenter and S. Grossberg, "The ART of adaptive pattern recognition by a self organizing neural network," *Computer*, pp. 77–87, 1988.

[5] Bart Kosko, "Bidirectional associative memories," *IEEE Trans. on System, Man and Cybernetics*, Vol. 18, No. 1, pp. 49–60, 1988.

[6] Bart Kosko, *Neural networks and fuzzy systems – a dynamical systems approach to machine intelligence*, Prentice–Hall, 1992.

[7] Y. F. Wang, J. B. Cruz, Jr. and J. H. Mulligan, "Two codig strategies for bidirectional associative memory," *IEEE Trans. on Neural Networks*, Vol. 1, No. 1, pp. 181–192, 1990.

[8] Y. F. Wang, J. B. Cruz, Jr. and J. H. Mulligan, "Guaranteed recall of all training pairs for bidirectional associative memory," *IEEE Trans. on Neural Networks*, Vol. 2, No. 6, pp. 559–567, 1991.

[9] H. Kang, "Multilayer associative neural network (MANN's): Storage capacity versus perfect recall," *IEEE Trans. on Neural Networks*, Vol. 5, pp. 812–822, 1994.

[10] Z. Wang, "A bidirectional associative memory based on optimal linear associative meory," *IEEE Trans. on Computers*, Vol. 45, pp. 1171–1179, 1996.

[11] P. K. Simpson, "Higher–ordered and intraconnected bidirectional associative memories," *IEEE Trans. on System, Man and Cybernetics*, Vol. 20, No. 3, pp. 637–652, 1990.

[12] X. Zhuang, Y. Huang and S.–S. Chen, "Better learning for bidirectional associative memory," *Neural Networks*, Vol. 6, No. 8, pp. 1131-1146, 1993.

[13] H. Oh and S. C. Kothari, "Adaptation of the relaxation method for learning in bidirectional associative memory," *IEEE Trans. on Neural Networks*, Vol. 5, pp. 573–583, 1994.

[14] C.–C. Wang and H.–S. Don, "An analysis of high–capacity discrete exponential BAM," *IEEE Trans. on Neural Networks*, Vol. 6, No. 2, pp. 492–496, 1995.

[15] Zong-Ben Xu, Yee Leung, and Xiang-Wei He, "Asymmetric bidirectional associative memories," *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 24, No. 10, pp. 1558–1563, 1994.

[16] Hongchi Shi, Yunxin Zhao, and Xinhua Zhuang, "A general model for bidirectional associative memories" *IEEE Trans. on Systems, Man and Cybernetics–Part B: Cybernetics*, Vol. 28, No. 4, pp. 511–518, 1998.

[17] M. Hagiwara, "Multidimensional associative memory", in *Proceedings of International Joint Conference on Neural Networks*, Vol. I, pp. 3–6, Washington DC, 1990.

[18] C. Choi and J. J. Lee, "Finding multiple local minima using chaotic jump," *International Journal of Cooperative Information Systems*, Vol. 7, No. 1, pp. 105–115, 1998.

[19] Wulfram Gerstner, Andreas K. Kreiter, Henry Markram, and Andreas V.M. Herz, "Neural codes: firing rates and beyond," *Proc. Natl. Acad. Sci.*, Vol. 94, pp. 12740–12741, 1997.

[20] John J. Hopfield and Andreas V.M. Herz, "Rapid local synchronization of action potentials: Toward computation with coupled integrate–and–fire neurons," *Proc. Natl. Acad. Sci.*, Vol. 92, pp. 6655–6662, 1995.

[21] Lipo Wang, "Learning and retrieving spatio–temporal sequences with any static associative neural network," *IEEE Trans. on Circuits and Systems–II: Analog and digital signal processing*, Vol. 45, No. 6, pp. 729–738, 1998.

TABLE I

ORIGINAL STORING PAIRS AND LEARNING PAIRS

	Original pairs	For learning w^1	For learning w^2
1	$x^\mu \leftrightarrow y^{\mu_1}$ $x^\mu \leftrightarrow y^{\mu_2}$	$x^\mu \rightarrow y^\mu$	$x^\mu \leftarrow y^{\mu_1}$ $x^\mu \leftarrow y^{\mu_2}$ $x^\mu \leftarrow y^\mu$
2	$x^{\mu_1} \leftrightarrow y^{\mu_1}$ $x^{\mu_1} \leftrightarrow y^{\mu_2}$ $x^{\mu_2} \leftrightarrow y^{\mu_2}$	$x^{\mu_1} \rightarrow y^\mu$ $x^{\mu_2} \rightarrow y^{\mu_2}$ $x^\mu \rightarrow y^\mu$	$x^{\mu_1} \leftarrow y^{\mu_1}$ $x^\mu \leftarrow y^{\mu_2}$ $x^\mu \leftarrow y^\mu$
3	$x^{\mu_1} \leftrightarrow y^{\mu_1}$ $x^{\mu_1} \leftrightarrow y^{\mu_2}$ $x^{\mu_2} \leftrightarrow y^{\mu_2}$ $x^{\mu_2} \leftrightarrow y^{\mu_2}$	$x^{\mu_1} \rightarrow y^\mu$ $x^{\mu_2} \rightarrow y^{\mu_2}$ $x^\mu \rightarrow y^\mu$	$x^{\mu_1} \leftarrow y^{\mu_1}$ $x^\mu \leftarrow y^{\mu_2}$ $x^\mu \leftarrow y^\mu$

TABLE II

RETRIEVING SEQUENCES

Case	Retrieving sequence
1	$x^\mu \leftrightarrow y^\mu \rightarrow y^{\mu_1} \rightarrow y^{\mu_2} \cdots$
2	$\cdots x^{\mu_2} \leftarrow x^{\mu_1} \leftarrow x^\mu \leftrightarrow y^\mu \rightarrow y^{\mu_1} \rightarrow y^{\mu_2} \cdots$
3	$\cdots x^{\mu_2} \leftarrow x^{\mu_1} \leftarrow x^\mu \leftrightarrow y^\mu \rightarrow y^{\mu_1} \rightarrow y^{\mu_2} \cdots$

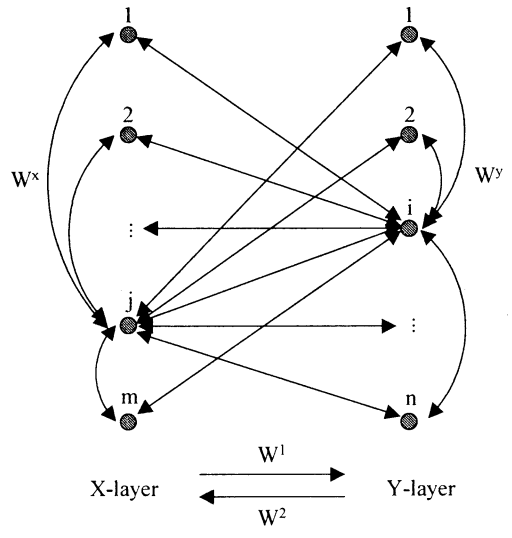


Fig. 1. Dynamic bidirectional associative memory structure

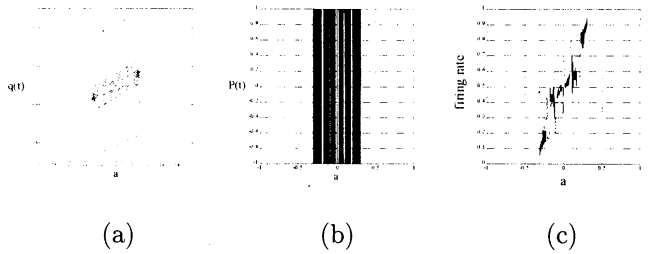


Fig. 2. Characteristics of chaotic neuron with parameters, $k = 0.2$, $\alpha = 0.5$, $\sigma = 0.125$: (a) internal state, $q(t)$; (b) output, $p(t)$; and (c) firing rate of $p(t)$.

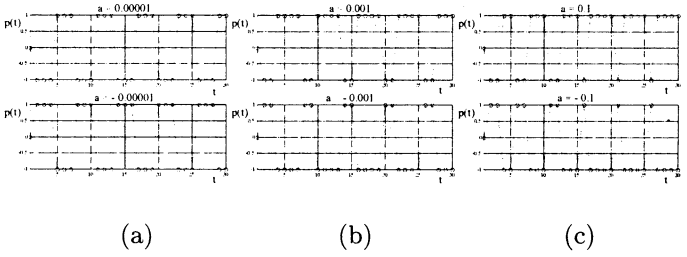


Fig. 3. Periodic orbits of chaotic neuron: (a) period–6 orbit with a firing rate of 1/2 (1/2) for $a = \pm 0.00001$; (b) period–6 orbit with a firing rate of 2/3 (1/3) for $a = \pm 0.001$; and (c) period–5 orbit with a firing rate of 4/5 (1/5) for $a = \pm 0.1$;

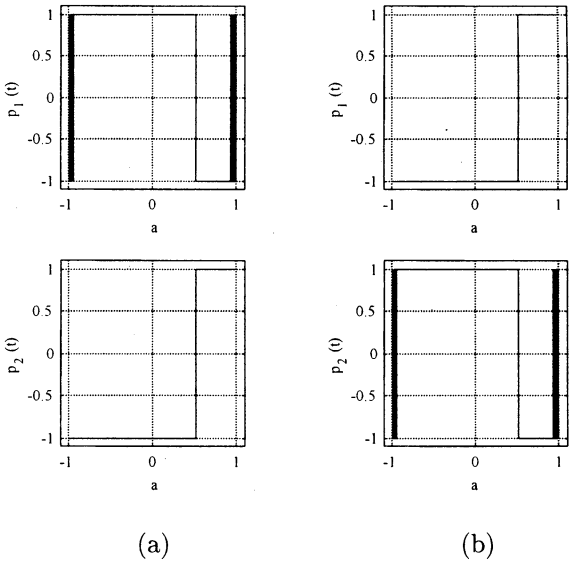


Fig. 4. Coupling effect between two chaotic neurons: (a) $p_1(t) = 1$ and $p_2(t) = -1$ when $w_{12}^y = -1.0$ and $w_{21}^y = -0.6$; and (b) $p_1(t) = -1$ and $p_2(t) = 1$ when $w_{12}^y = -0.6$ and $w_{21}^y = -1.0$;

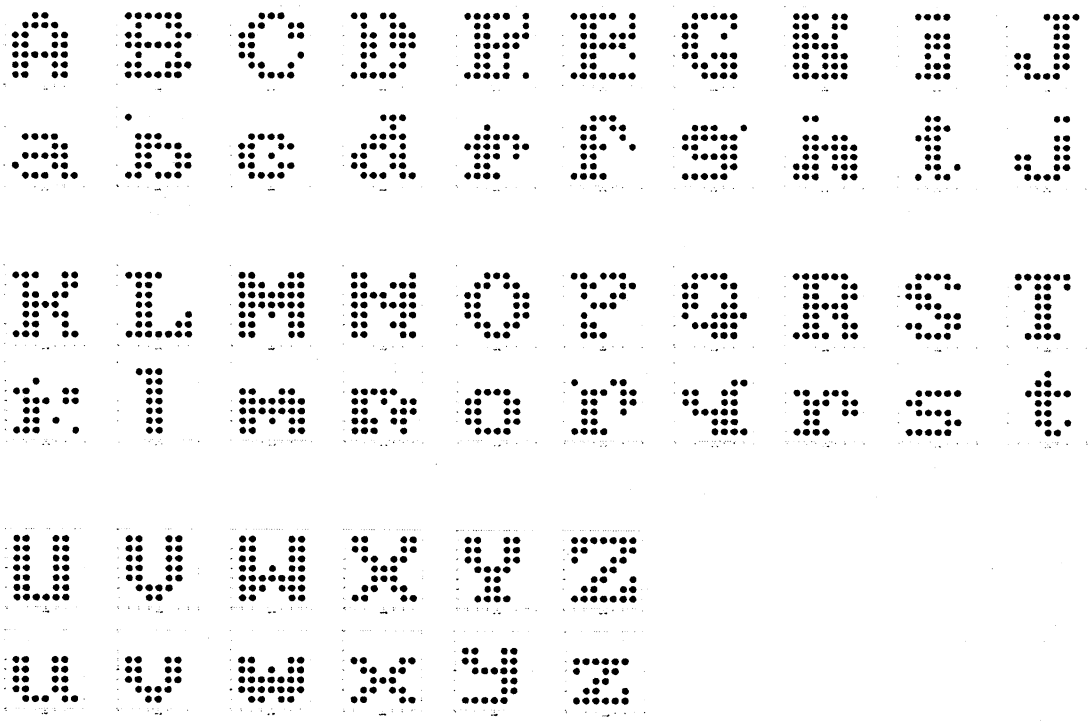


Fig. 5. Recalled letter pairs from 8% noise corrupted uppercase letters.

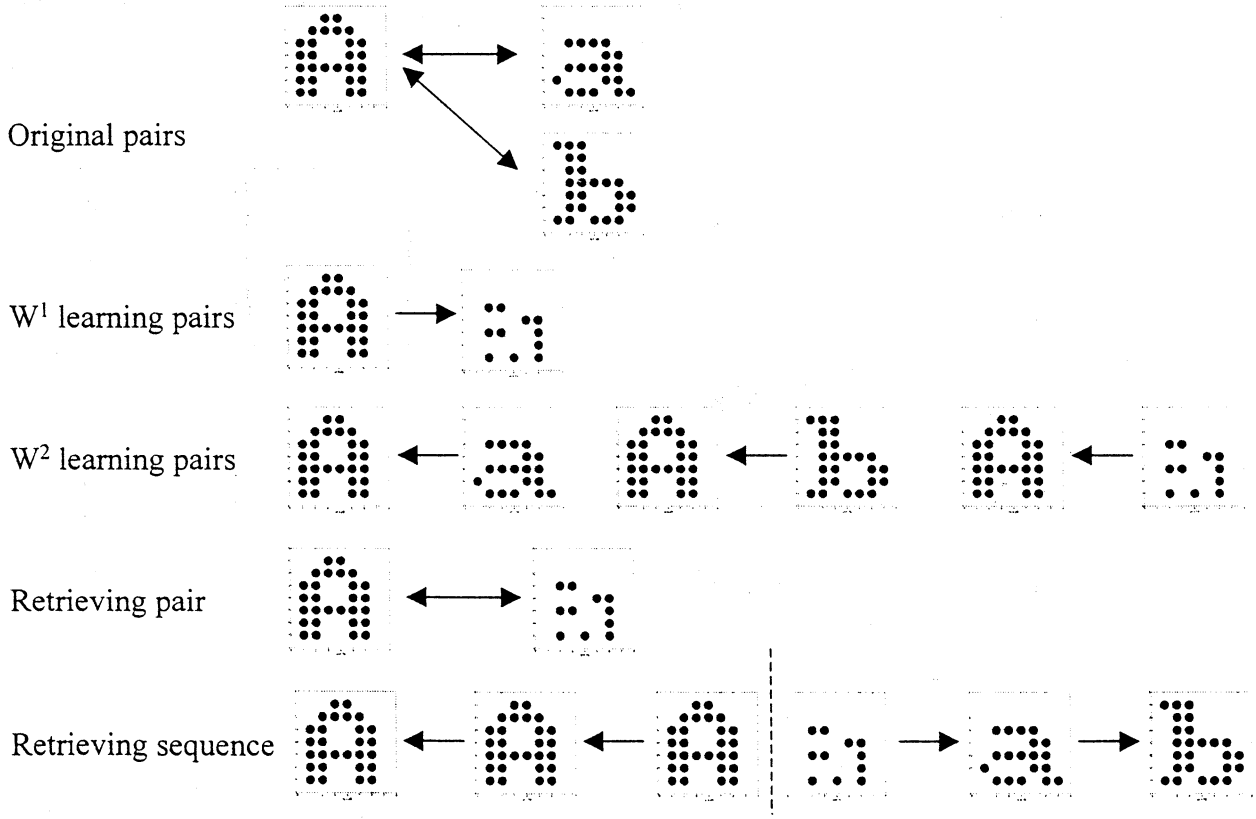


Fig. 6. Dynamic bidirectional multiple association: case 1.

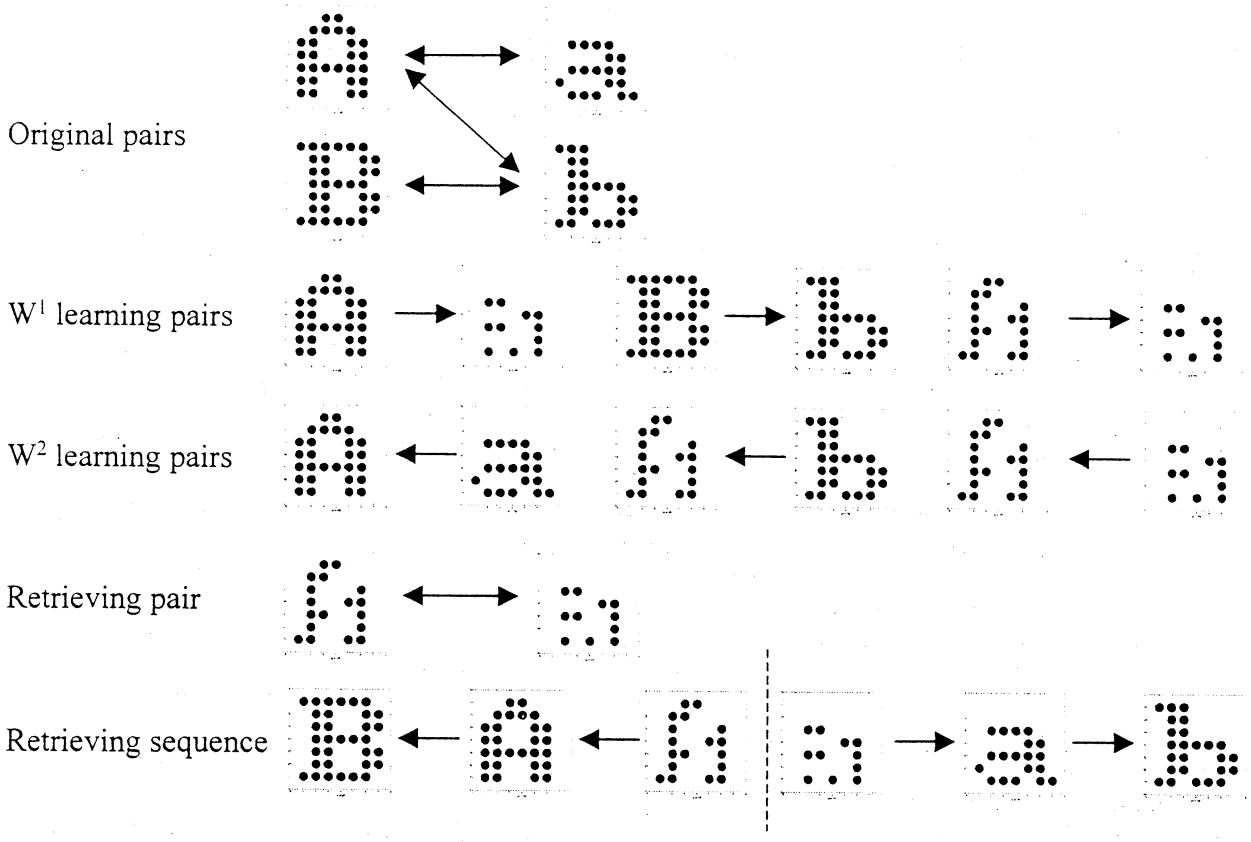


Fig. 7. Dynamic bidirectional multiple association: case 2.

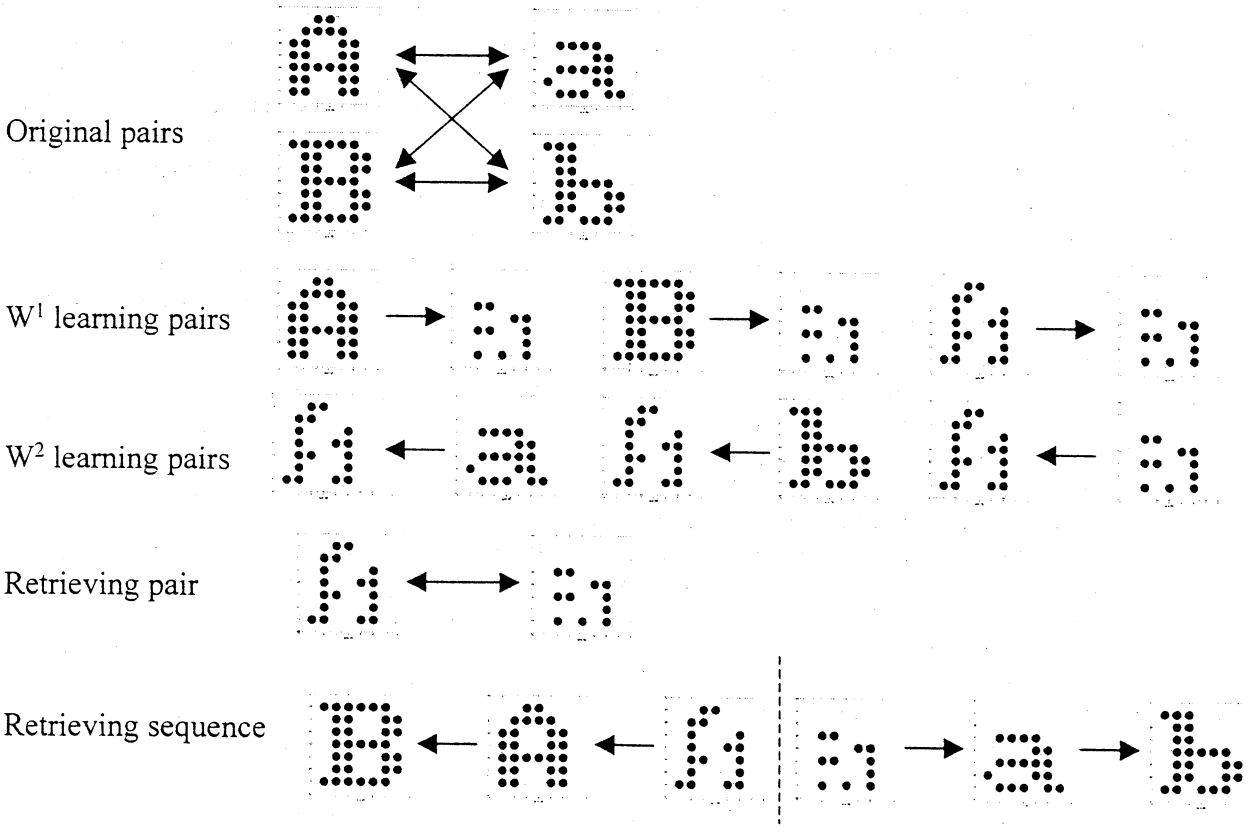


Fig. 8. Dynamic bidirectional multiple association: case 3.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Bottom-Up Way to Develop Multi-cellular Digital Organisms

Y.G. ZHANG
Institute of System Science
Academia Sinica
Beijing, P.R.China, 100080

M. SUGISAKA
Dept. of E.&E Engineering
Oita university
Oita, Japan, 870-11

Xiaoqiang WU
Institute of System Science
Academia Sinica
Beijing, P.R.China,100080

ABSTRACT

How multi-cellular creatures develop from single cells into multi-cellular forms is a basic question for the research of AL. In this paper, we propose a possible answer to that question by developing multi-cellular digital organisms from singe-cell digital organisms. We have done a experiment on computer, first of all we define a model of singe-cell organism which owns open-ended evolvability and self-replicating mechanism, then these single-cell digital organisms can develop into multi-cellular creatures which have better adaptability than single-cell ones under certain environments, Phenomena of cell differentiation and cell self-organizing are observed during the development of such multi-cellular digital organisms.

Keywords: single-cell unit, multi-cell digital organization, Metabolism, Reproduction, Bifurcation

I. INTRODUCTION

In the research of Artificial Life some one pay attention to the whole body which is lifelike, for example, the artificial fish and evolutionary robots; some one more emphasize the unit which has artificial life, for example, the Metabolism-Repair system. If you could realize that the society and the economy system in a country is also lifelike and it has the characteristic of life, it will be a huge artificial life system. So, there are many kind way to do the research on the Artificial Life. If we make a order to artificial life systems by the size, the M-R system is at the bottom, the artificial fish and evolutionary robots is in the middle and the economy system is on the top. We suppose the M-R system is **a single-cell artificial life unit**, and try to organize a **multi-cell digital organism** which is consisted of single-cell unit and has more complexity than it. This is the purpose of our research in this paper. In deed, the development of multi-cellular organisms is a very complex process. Several models have been proposed to describe it. Earlier models are L-system and Metabolism-Repair system[2,3]. We have dropped a modified model of M-R system to describe a single-cell units[1,2,3]. Recent works belong to Ray and Kitano. Ray[4] pointed out in his paper that the development of multi-cellular digital organisms should have the following features: 1) multi-cellular organisms originate as single cells, which develop into multi-celled forms through a process of binary cell splitting; 2) each cell of a multi-celled individual has the same genetic material as the original cell from which the whole developed; 3) the different cells of the fully developed form have the potential for differentiation, in the sense that they can express different parts of the gene. Kitano proposed that the development of multi-cellular organisms can be regarded as an emergent phenomenon that raised in the evolution of metabolism[5].

In this paper, we propose a bottom-up way to develop multi-cellular digital organisms. A model of metabolism in cells is carefully defined. Based on the model, we describe the mechanism of cell splitting, cell death, intercellular communication, cell differentiation and self-organizing of cells. The experiment results is given at the end of this paper, which shows that our model can really explain the phenomena that happen in the development of multi-cellular organisms.

II. STRUCTURE OF THE SYSTEM AND DYNAMICS

In our system we use the biological terminology **cell** and **organization** to describe the components in our system. Cell is at bottom level and Organization is the above level. First all, we define the unit, single-cell, and then to grow up a organization from the units by some lifelike-rules, that is the meaning of bottom-up way.

The single-cell units can be thought as any digital units which has basic process **Metabolism** and a information process to read and explain the **gene,** which is a way to store the whole control rules. There are some complex dynamics to make a multi-cell digital organization from the single-cell units. The dynamics includes rules as the following:

Metabolism--This is a making new product process and matter exchange process between the inside of unit and the environment outside.

Reproduction--This is a process of copying-itself, usually it is a splitting process from a unit.

Death--It is a process to keep the multi-cell digit organization growing and balance the processes of Metabolism and Reproduction.

These processes M-R-D consist of the basic and the simplest dynamics for the development of a multi-cell digital organization. To achieve more complex development the bifurcation process is necessary.

Bifurcation--It is a process to produce two different unit which has the same gene and in a same individual. This process is very essential for developing more complex multi-cell digital organization.

All of these processes are controlled by many rules, all these rules are encoded in a same way. We call these kind of rules encoded to control and organizing a multi-cell developing as **gene.** In the above M-R-D-B process the gene are decoded. This kind of control is quite different from the usual control. The matter of fact is that usual control is from outside of system, however, the gene control is hidden inside of system.
We have done some experiment for our idea, it is showed in late section.

III. DIVERSITY AND COMPLEXITY FOR DIGITAL LIFE

Similar to the real biological word, the artificial digital creatures have their diversity and complexity. In fact, There are some stochastic fluctuation in the M-R-D-B processes when they were controlled by gene, the mutation is the reason to result in diversity. Also, the single-cell units in different forms lead to diversity.
Complexity is a comparable concept, the single-cell system has less complexity in structure and dynamics than the multi-cell organization. Furthermore, a individual with multi organization has more complexity I dynamics and function and structure than the multi-cell organization. Both diversity and complexity are from the complexity of the gene. This means **evolution.** So, the grade of evolution has different complexity. More higher the grade of evolution is, more complexity the system has. If we can not create the multi-cell digital organization, then we can not make more complex digital life system. So, the research in this paper is a necessary step to realize the more complex system with lifelike characteristic.

IV. OUR EXPERIMENT

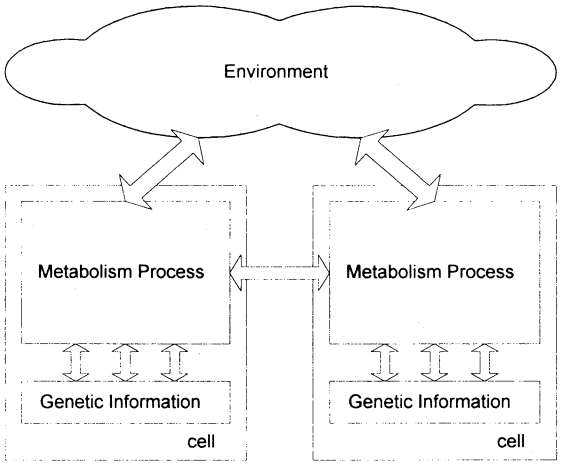


Fig. 1 Overview of our model

The Model

Fig. 1 shows the overview of our model. There exists matter exchange between the environment and cells. Cells import useful matter from their environment and the neighbors, export the product of metabolism into the environment or other cells. Through the process of matter exchange, cells acquire the energy and the information which are necessary for their proper development.

In the following subsections we will give the definition of the environment, the metabolism in cells and the genetic information in cells, then some phenomena that always happen in the development of multi-cellular organisms are described on the foundation of these definitions.

Environment

Environment is the place where the cells live. In the beginning, it contains some simple elements which sustain the fundamental metabolism of cells; later, when the metabolism of cells start, the environment will also contains the various matter produced by metabolism in cells. In the model, the diffusion of matter is simulated with the hypothesis that the speed of diffusion of some matter from one area to another area is proportional to the difference of concentration of that matter between the two areas.

Metabolism Process

Metabolism is mainly determined by Metabolism rules (M-rules). They determine whether a reactions will or not happen when different kinds of matter are put together and what kinds of matter will be produced by those reactions. **M-rules result in the possibility of the diversity of natural organisms.** The relation between M-rules and genetic information is that genetic information are the encoded M-rules to make itself expressed.



Fig. 2 format of chemical rules

As shown in Fig. 2, a M-rule consists of reactants and products. As suggested by Kitano, reactants and products are represented by binary strings with fixed length. The binary string of length n, representing reactants and products, means there are 2^n possible reactants and products in the model. So, the M-rules in the model can be listed by a $2^n x 2^n$ reaction matrix M with elements $m_{i,j}$ which is the product of the reaction between i , j . where $0 \le m_{i,j}$, i,j, $\le 2^n$. The reaction matrix should have features: 1) $\forall$ i, $0 \le i \le 2^n$, $m_{i,i}$ =void ,which means a reactant can not react with itself; 2)) $\forall$ i, j, $0 \le i, j \le 2^n$, $m_{i,j} = m_{j,i}$, which means the reaction will neglect the order between the two reactants; 3) if $m_{i,j} \neq$ avoid, then $m_{i-1,j}$, $m_{i+1,j}$, $m_{i,j-1}$, $m_{i,j+1}$, should not be equal to $m_{i,j}$; 4) $m_{i,j} \neq i,j$.
The definition of gene is a encoded rules. A piece of gene information consists of activating factors, inhibiting factors and a sequence of reactants(shown in Fig. 3).
Activating factors facilitate the expression of basic genetic information, inhibiting factors cumber it. Both activating factors and inhibiting factors are arrays of the matters in a

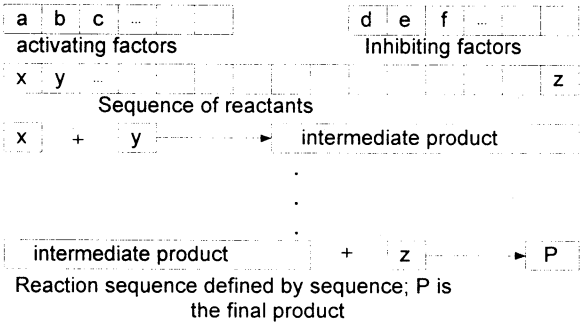


Fig. 3 format of genetic information

cell. The difference between the concentration of activating factors and that of inhibiting factor controls whether the piece of genetic information can be expressed or not and the speed of expression. The sequence of reactants determine the sequence of reactions among the reactants. Different sequences will result in different products. In the model, two arbitrary matters not must react with each other freely, even the certain reaction between the two matters has been defined in the reaction matrix M. Only when they are included in some piece of gene information and the expressing of that piece of gene information is activated, can the reactions take place. This assumption ensures that the metabolism is controlled by gene in cells. The expressing of the gene will stop in case of two conditions: 1) anticipate reactions are not defined in the reaction matrix M; 2) required reactants are not available.

Matter Exchange

The matter exchange between cells and the environment is the foundation of metabolism in cells, There are two kinds of matter exchange between cells and the environment: **active transport and diffusion**. In the model, some matters can enter or leave the cell by diffusion, others however depend on certain matters to transport them actively.

$$Speed\ of\ Diffuse_i^m(t) = D\left[X^m(t) - x_i^m\right.$$

$$Speed\ of\ Active\ TransIn_i^m(t) = I \cdot X^m(t) \cdot x_i^{m'}(t)$$

$$Speed\ of\ Active\ TransOut_i^m(t) = O \cdot x^m(t) \cdot x_i^{m''}(t)$$

Here, X^m represents the concentration of matter m in the environment; x_i^m, $x_i^{m'}$, $x_i^{m''}$ represent the concentration of matter m, m' (the matter which transport m into cells) and m'' (the matter which transport m out of cells) in the i-th cell respectively; D, I, O are constants.

As shown in the previous equations, the speed of diffusion of some matter is proportional to the difference of the concentration between the chemical inside the cells and outside the cells. The speed of active transportation of some chemical is proportional to the concentration of the matter and its transporters in cells.

The matter exchange among cells is a little different from the matter exchange between cells and the environment. There also exist diffusion among cells. When two cells are leaning against each other, there have been the possibility to exchange matters. To actively transport some matter m from one cell to another cell, there must exist certain matter m''' in both cells. The speed of active transportation of some matter between two cells is proportional to the concentration of the matter and its transporters in both cells.

Matter exchange is very important for the correct development of multi-cellular organisms. It is also the basis for intercellular communication and self-organizing of cells. Detailed analysis will be given in next section.

Reproduction Process

Reproduction process in a Cell basically is split. In the model, we consider that cell splitting is relative to the concentration of a particular chemical div. When x_i^{div} is large than a given value, the i-th cell begins the process of splitting.

During the process of cell splitting, a series of events will take place in the cells. First, the old cell will replicate its gene information and mutation may occur with a certain probability which make the cell own open-ended evolvability. Then, the matters in the old cell may also be replicated and distributed into the new cells randomly. The fluctuation in the process of gene allocation may result in that one of the new cells owns more gene information than the other. The chemical div is consumed out in the process of cell splitting.

Death Process

There exist two kinds of death in the model. One is unnatural death which is caused by the lack of some matters or the abrupt change of environment. This kind of death is often accompanied with the lower and lower level of metabolism. Another kind of death is programmed cell death which is caused by the increment of the concentration of some chemical. We call this as chemical *death*. When x_i^{death} reaches a given value, the i-th cell will be disassembled and dead.

When a gene mutation destroys the gene information which control the production of chemical *death* , it will result in the unrestricted development of cells which is similar to cancer and will destroy other cells ultimately.

During the process of cell splitting, the chemical *death* is scattered into new cells randomly.

The programmed cell death is a necessary condition of proper development of multi-cellular organisms.

Further Analysis

In this section, we will use Petri net to analyze the model that we proposed.

Petri net is often used to model the parallel systems in theoretic computer science. The Petri net that we use here is called place-transition net.

A place-transition net consists of

* places, represented as circles (○);
* transitions, represented as boxes (□);
* arrows from places to transitions ○→□;
* arrows from transitions to places □→○;
* a capacity indication for every place(represented as label K = ...);
* a weight for every arrow(represented as a number);
* an initial marking, defining the initial number of tokens for every place(cannot be greater than the indicated capacity).

Further knowledge about Petri net can be found in [6,7].

The gene information we defined in previous section can be model by Petri net, as shown in Fig. 4. The metabolism in cells can also be described by a Perti net in which the existence or absence of matters correspond to the places and the M-rules correspond to the transitions. Given the gene information and initial distribution of matters in a cell, we can use Petri nets to analyze the performance of the cell and determine what gene information are pivotal for the cell. This is very important for distinguishing the different cells which develop from the original cell.

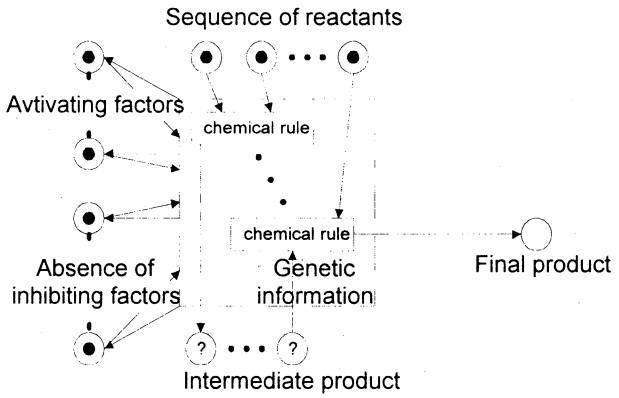


Fig. 4 The Petri net for a piece of genetic information

Results of Experiments

In the experiment, the length of the binary strings which represent the matters is 4, which means there are sixteen matters in the model at most. Table. 1 is the reaction matrix M. There mainly exists matter 0 in the environment. In cells, there are only one activating factor and one inhibiting factor in each piece of gene information and there are two reactants in the sequence of reactants. The amount of each matter has limitation in cells. We arbitrarily designate chemical 15 as *death*. If the amount of *death* exceeds thirty in a cell, the cell will die. The cell will also die if there are not gene expression in the cell for a certain time. The chemical *div* is designated as chemical 4. If the amount of death exceeds eight, the cell will split. The rules of the active matter exchange among cells are defined in Table. 2. The rate of mutation is 0.05 in cells.

We first designed an original cell which owns gene information and initial matters distribution. Then, several copies of the cell are put into the digital environment to develop and evolve. Through several generations, only the cells which adapted the environment best could survive.

Through analyzing the metabolism of the cells by drawing their correspondent Petri nets, we find one the evolved cells has very interesting features. It first depends on gene information 4 to produce matter 2 which is crucial for the cell to produce *div* and survive. Later, after splitting several times, the cells begin to form a group. The inner cells show different behaviors with the outer cell, they do not depend on gene information 4 to produce matter 2 any longer, but depend on gene information 9, and both the relationship among the outer cells and the relationship among the inner cells increased. We interpret this as cell differentiation and cell self-organizing. We think that the cell differentiation is caused by the deficiency of matter 0 in the inner cells. The snapshots of its development are shown in Fig. 5 and its gene information is listed in Table. 3.

Conclusions

In this paper, we propose a new framework for developing multi-cellular digital organism. The metabolism process of cells are defined. Some cell activities, such as cell splitting and cell death, are modeled. Petri net is proposed as a tool to analyze the metabolism of cells. The result of experiments proves that our model can explain the phenomena of cell differentiation and cell self-organizing that take place in the development of multi-cellular organisms. Our further work will focus on improving definition of chemical rules and theoretic analysis of the model.

References

[1] Zhang,Y.G., Sugisaka,M., Li,X.J., Lifelike artificial tree based on the growth IFS, *Applied Mathematics and Computation, Vol.91, No.1, pp.3-8,* 1998

[2] Zhang,Y.G., Sugisaka,M., Some new look on the Metabolism-Repair system, *The proceedings of AROB97*

[3] Zhang,Y.G., Sugisaka,M., Xu,C.M., Modification of M-R system and its simulation, *ARTIFICIAL LIFE AND ROBOTICS, Vol. 1, No.2,* 1998

[4] Thearling, K., Ray, T. S., "Evolving Multi-cellular Artificial Life," *Artificial Life VI*: 283-288. 1994.

[5] Kitano, H., "Evolution of Metabolism for Morphogenesis," *Artificial Life VI*: 49-58. 1994.

[6] Reutenauer, C., "The Mathematics of Petri Nets," Masson and Prentice Hall International (UK) Ltd. 1990.

[7] Reisig, W., "A Primer in Petri Net Design," Springer-Verlag. 1992.

Table. 1 Reaction matrix. In (a,b), a is the amount of the product; b is the product.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0		3,2	2,1													
1	3, 2		1,0													
2	2, 1	1,0					2,9		3,15	1,6						3,4
3							2,11				2,14					
4						3,6							1,3			3,5
5					3,6		3,7									
6			2,9	2,11		3,7		3,8		3,2						
7							3,8		1,2					3,8		
8			3,15					1,2								
9			1,6				3,2									
10				2,14									2,13		2,11	
11																
12					1,3						2,13			3,14		
13								3,8					3,14			
14											2,11					
15			3,4		3,5											

Table. 2 Matter transition rules

chemical	transport medium
0	none
1	7
2	11
4	13
5	14
6	8
7	12
8	none
9	5
10	none
11	none
12	none

Table. 3 The genetic information of a cell

No.	Activating Factors	Inhibiting Factors	The Sequence of Reactants
1			2,15
2		13	1,2
3			0,2
4			0,1
5	1		15,4
6			4,5
7	9		6,2
8			6,7
9		0	6,9
10			8,2
11	12		7,8
12			9,2
13			6,5

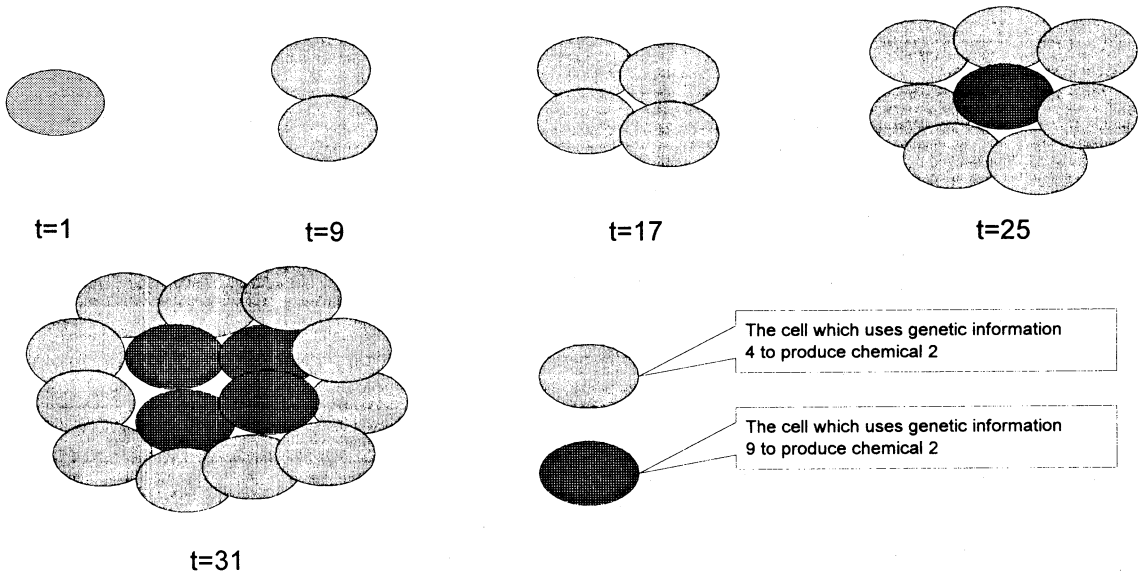


Fig. 5 The snapshots of the development of a cell

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Human-Machine Cooperative System for Natural Language Processing

Yue Fei, Ruwei Dai

Institute of Automation, Chinese Academy of Science
Beijing 100080, P. R. China
fei_yue@163.net, dai@ht.rol.cn.net

ABSTRACT

A human-machine cooperative system for natural language processing is proposed. First, the system uses extended competitive supervised learning algorithm to form the distributed representations, which could serve as the basis of communication among the system modules. Second, a human-machine interactive mechanism is integrated into the system to enhance the ability for new words processing. Because of human-machine cooperation, the system could simulate higher lever cognitive behavior. Finally, this human-machine cooperation is also used in a computer art system for antithesis generation, and gives out satisfactory result.

Keywords: Human-machine cooperation, Extended competitive supervised learning, Natural language processing.

1. Introduction

Now it comes to an era of the tremendous explosion in the amount of information available. As a result, it's urgent to transform the field of Natural Language Processing (NLP) from theoretical models of very specific linguistic phenomena to one guided by computational models that account for a wide variety of phenomena that occur in real-world text. However, compared to other domains of Artificial Intelligence, the progress of NLP has been slow. A reason explained by Minsky[1] is that none of our system can use "common sense" -- computers have no access to the meanings of most ordinary words and phrases. Then it leads to the fact that systems even modules in a system cannot communicate with each other.

In the kinds of models for NLP, the major appeal for connectionism is that the processing knowledge can be extracted from examples through training process. Another feature of these connectionist models, which maybe its advantage as well as disadvantage, is that they generally have very little internal architectural complexity. With this feature, the neural network may appropriate for modeling low-level tasks. While it meets some difficulty in simulating more complex cognitive phenomena. However, one of plausible solutions proposed in Dai[2] is to integrate several network modules and the role of human into a whole system. Then, it also brings out the problem of the communication between modules and human. In any integrated system, it needs a common set of terms to serve as the basis of intercommunication. In the system for NLP, this set of terms could be the distributed semantic representation, through which, modules of the system could be trained separately and then communicate with each other to accomplish high-level cognitive tasks.

In this paper, the approach, Extended Competitive Learning (ECSL), intending to solve the communication problem in connectionist systems, is enlightened by the idea of human-machine cooperation. It is no doubts that the issue of modeling natural language belongs to the ill-structured problems, which means hard to construct models for these problems. However, building models with human-machine cooperative method is a good solution for this kind of problems. In the following section, we begin with the extended supervised learning algorithm to develop the distributed representation and propose a human-machine interactive mechanism to deal with the new word processing. The ECSL is also applied to the antithesis generation system.

2. Extended Competitive Learning Algorithm

2.1 the architecture of the ECSL

In real-world text, one word may have several different meanings; therefore it is better to use separate representation to stand for different meaning. The proposed Extended Competitive Supervised Learning aims particularly at solving the ambiguity of words. The work owes much to the FGREP proposed by Miikkulainen & Dyer[3] and the competitive learning algorithm by Wang et al.[4] and Fei et al.[5], but we propose an alternative architecture to develop separate distributed representation for ambiguous words.

The architecture of ECSL is similar to a three-layer backward propagation network (Figure 1). The input layer is divided into several groups and each group is divided into several assemblies, which represent different meanings of the same word. The lexicon contents all the distributed representations of the word, and the different representations of the same word are specified by a common ID. As a result, it is easy to tell apart the similar meaning of different word. The lexicon also forms the target representation for the output layer. The error signal generated from the output layer propagates through the hidden layer to the input layer. At the input layer, the different assemblies of the ambiguous word

develop new representations through the competitive supervised learning.

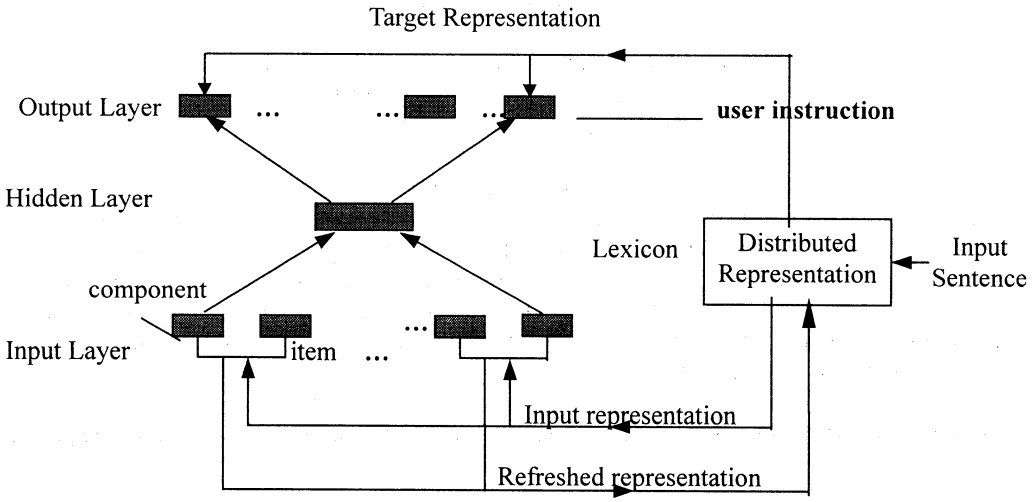


Figure 1. The snapshot of the ECSL architecture

2.2 ECSL Algorithm

The key point of the ECSL is to separate different meaning of the ambiguous words. This process happens at the input layer while the error signal propagates to the input layer. The assemblies can be changed to develop the different representations.

At the beginning of the training process, the lexicon randomizes every representation of the words, so it contents no pre-encoded information. Every representation of the ambiguous word is input to the first layer. At the output layer, the network is intended to produce the correct meaning. At every step, the network compares the output result to all the representations of the ambiguous word, and selects the most similar representation as the target representation to modify at the input layer. After several epochs, the representations of the ambiguous word could be distinguished.

In a sense, the process to modify the representation could be taken as a kind of competitive behavior among the different representations of the same word. The following is the extended competitive learning algorithm.

Step 1. Initialize all the representations in the lexicon to uniformly distribute in the interval [0,1]; And the weights within [-1, 1];

Step 2. Change the input sentence into semantic representation and concatenate all the representations to the input layer.

Step 3. Forward compute. From all the representations of item c, select the representation, R_{cmax}, which is most similar to the output result, as the target representation to form the error signal:

$$E = \frac{1}{2}\sum_{ci}(R_{ci} - R_{c\max i})^2 \tag{1}$$

Back propagate the error signal to the input layer according to the BP algorithm (Rumelhart et al., 1986):

$$\delta_{1i} = \sum_{j} \delta_{2j} w_{1ij} \tag{2}$$

Where δ_{ij} stands for the error signal for unit j in layer i, and w_{1ij} is the weight between unit i in the input layer and unit j of the hidden layer.

Step 4. The representations at the input layer are changed according to the following error function:

$$\Delta R_{cki} = \eta \delta_{1i} \tag{3}$$

Where R_{cki} is the kth representation component i of the item c. δ_{1i} is the error signal of the corresponding input layer unit and η is the learning rate. To assure the limit of the representation value, the new component is obtained as

$$R_{cki}(t+1) = \lambda \, max[-1, min[1, R_{cki}(t) + \Delta R_{cki}]] \tag{4}$$

Where λ is computed according to the following equation:

$$\lambda = \begin{cases} 1, & R_{cki} \in R_{max} \\ 0, & R_{cki} \notin R_{max} \end{cases}. \tag{5}$$

Step 5. If there are training samples left, go back to step 2. Otherwise, go on to step 6.

Step 6. If the error is smaller than the desired value, training is finished. Otherwise, return to step 2 to begin another training epoch.

The representation at the input layer could be seen as an extra layer of weights. The adjustment of representations mainly takes place in the corresponding connections, while others are changed only by the momentum term. The training process is a kind of competitive course, in which the maximal output assembly inactivates other ones representing the same item. The process for the ECSL network to develop representations is based on the credit that learning a language is learning the use of the language elements. Since the similar sentence structure may be met in the training sample set, the ECSL has the ability to expect the new word's meaning by the given context.

2.3 Human-machine interactive mechanism in ECSL

The human-machine interactive mechanism integrated into the ECSL is to enhance its ability for new words processing. As we said above, the ECSL has the ability to create the expectation for the new words. However, since every word has its own distinct way in practical using while the sample set for training the ECSL network is limited, the expectation of the new word maybe incorrect. It also could meet this case, the new representation generated by the network and the learned words in the same category could be too similar to discern the new word's own feature. To avoid the inaccuracy and similarity, we need extra supervise to guide the expectation of new words. In another case, the new word maybe ambiguous, the supervisor could tell the system to set multiple initial representation for it.

In ECSL, the human-machine cooperative mechanism is to ask the user yes/no question about the new words, until the network could find a proper representation for the new words. In a sense, it is a kind of supervised learning, and the learning occurs during testing or using. Both the machine and human encode the distributed representations. Consequently, the lexicon is not unchangeable, but could enlarge its vocabulary as the system is under the instruction of the user.

It is reasonable to question that how the ECSL could find the proper representation through the limited several answers. Indeed, the ECSL does not only learn the new concept by the answers given by the user, it develops the new representation based on the expectation. The detail process will be discussed in the ECSL's application in the antithesis generation system.

4. Application in the Antithesis Generation System

The ECSL could be used in many practical systems, such as question-answering and immediate translation. In our projects, we use it in a computer art system for antithesis generation. Antithesis, also called couplets in English, is a kind of popular Chinese literature. Antithesis is unique for its succinct but vivid description and its strict format. One antithesis often consists of two sentences: one is called up-sentence, and another down-sentence. The two sentences are required to be similar or flowing in meaning and identical in structure. In English, there also exist some sentence alike, but these couplets in English are far less strict, abundant and varied than Chinese antithesis. For example:

Easy come easy go.
It was the spring of hope; it was the winter of despair.
Joy for his fortune; honor for his valor.

In this computer art system, for a given up-sentence, it should output the down-sentence, which is matching to the given sentence according to the given rules. Many words in the antithesis have multiple meaning and usage. It is no doubt that this is a tough problem and requires the system should integrate at least two modules dealing with the sentence processing and sentence generation. However, the antithesis is also an idea area for connectionism, since many idioms and fixed couplets form some implicit rules, such as 青山（mountain） vs. 绿水(river). So we choose the neural networks to design the whole system, and the ECSL not only plays the role of communication basis, but also is a module of sentence processing, which sign the case role for sentences. In the system, based on the distributed representation lexicon, all the modules can be further trained to complete particular tasks. The following example illustrates how the ECSL works in the system.

In the Chinese antithesis, "春" (spring) belongs to the words that often use. But in Chinese, it also has two meaning: one means season, which is often used, while another means spring-like and could be used as a verb. For example:

春回大地（Spring comes.） (1)

Here *spring* (春) serves as a noun.

祖国万年春 （My homeland is forever spring-like.） (2)

Here *spring* （春） serves as a verb in this sentence.

The second meaning of the word 春(*spring*)seldom occurred in the antithesis. But the supervisor knows the word is an ambiguous one, so the network initializes two semantic representations for the word. For the seldom-used one, the ECSL network could expect that *spring* means *spring-like* and functions as a verb in the sentence according to many learned example structure similar to this case. For example:

山河千秋丽 (The land of my country is always beautiful.) (3)

The sentence 3 is the up-sentence of sentence 2. (There are also many sentences alike that are not couplets of the sentence 2.) At here, 丽 means beautiful and could function as a verb. So the ECSL network may expect 春 also could serve as a verb in the similar sentences. And now the human-machine mechanism is working. The network could generate the sentence:

山河千秋春 (4)

This sentence is correct in its meaning. So the supervisor gives *YES* as the answer and this signal the network to produce another meaning of 春 that similar to the representation of 丽. But the word 丽 also have other usage, for a instance, it could means *nice* in the sentence 风和日丽 (The breeze is warm and the sunshine *is nice.*). So the network could also expect 春 has similar usage. It outputs the sentence:

风和日春 (5)

And now the supervisor would not agree this sentence, so the *NO* signal gives to the network. According to this signal, the representation of 春 could be correct not too similar to the representation of 丽. After several learning epoch like this, the network could find the proper representation for the ambiguous word 春. One thing we must point out is that in the training set of antithesis, there are many sentences have similar structure as that of the sentence 2. So the outputs of the ECSL network may contain many expectations of one word. The criteria we use here is that for all the sample sentence containing the ambiguous word, the network would point out its proper representations.

5. Conclusion

In this paper, we propose a human-machine cooperative approach for natural language processing, which is aimed for developing semantic representation and solving ambiguity of the words. Because the system involves the active role of human, it could use small training set to deal with larger real-world text. Besides this advantage, the ECSL also provides an open semantic lexicon to serve as the basis for communication among modules in the integrated system. During the training and the using process, because human is the system's supervisor, the lexicon could be changed larger and more suitable for the particular problem intended to resolve. After integrated several modules and the human role, the connectionist system could simulate high-level cognitive phenomena such as antithesis understanding and generation.

According to the ECSL's application in the computer art system -- the antithesis generation system, the human-machine cooperative approach seems effective to those ill- structured problems in the domain of artificial intelligence. The human and the machine work together to model the language lexicon, which is hard to be represented by mathematics functions and computer programs. With this open lexicon, the system could be added more modules and without slowing down the learning process and increasing the complexity of computation. The system could also begin with a small training set, and gradually enlarge its vocabulary through the supervisor role of human user. We believe that human-machine cooperation is a potential approach to solve more complex natural language processing problem.

REFERENCES

1.Minsky M(1992), Future of AI technology. *Toshiba Review*. July 47(7), 1992

2.Dai R(1991), Meta-synthetic approach from qualitative to quantitative(in Chinese). PRAI 4(1):5-10(Chinese edition).

3.Miikkulainen, R. & Dyer, M. G. (1991), Natural language processing with modular PDP networks and distributed lexicon. *Cognitive Science.* 1991, 15:343-399.

4.Wang L, Fei Y, Dai R(1997), A new model of feedforward neural networks in pattern recognition -- competitive multilayer perceptrons and competitive supervised learning (in Chinese). PRAI 1997, 10(3): 186-195.

5.Fei Y, Wang L, Dai R(1998), The application of the competitive supervised learning in the pattern recognition system (in Chinese). Accepted by the *Journal of Automation* (Chinese edition).

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

An Adaptive Classifier System Tree for Extending Genetic Based Machine Learning in Dynamic Environment

Rui Jiang Yupin Luo Dongcheng Hu

Dept. of Automation, Tsinghua University,
Beijing 100084, P.R.China
EMail: { rjiang, luo }@mail.au.tsinghua.edu.cn

Abstract – **Autonomous agent should possess the ability of adapting its cognition structure to dynamic changing environment. This ability may be achieved when autonomous agents interact with the environment. An adaptive classifier system tree is proposed in this paper for extending genetic based machine learning in dynamic environment. The architecture has the properties of self-similarity and self-organization. When environment changing is inspected, autonomous agent can adapt its cognition structure to the new environment so that cognition can be achieved by a high efficiency. After a description of the dynamic structure and the principle of the structure's self-organization, experiments illustrate how the architecture works together with discussions are given.**

Keywords – **autonomous agents, genetic based machine learning, self-organization.**

1 Introduction

The traditional knowledge-based approach to artificial intelligence explains the cognitive abilities of the brain by means of symbol manipulation and reasoning. Although this approach is successfully applied in domains such as medical diagnosis and ore exploration [Buchanan[1]], it seems to lack the flexibility and expressiveness of natural cognitive systems. Much work done in behavior based robotics show it may be a better way to achieve this kind of cognition. [Holland[2,3]].

Early work done in behavior-based robotics focuses on the design of appropriate robot behavior and behavior coordination techniques [Brooks[4]]. Recent work done by M.Dorigo et al[5] develops an architecture of cognition based on both ethnologic and evolutionary considerations. Their work shows that the introduction of evolutionary approach to cognitive process is a plausible and powerful way to develop intelligent systems.

We point out, however, that an autonomous agent must possess the ability of adapting its own cognition structure to the changing environment. In this paper, we intend to construct an adaptive architecture of cognition based on this consideration. In this architecture, complex environment input can be inspected and divided into simple ones; simple cognition units are designed to achieve the cognition of these simple inputs and pass cognition result to higher level unit. After coordination by higher level unit, agent's final cognition result is obtained. The architecture has the properties of self-similarity and self-organization.

The rest of the paper is organized as follows: In Section II, we briefly review principles of genetic algorithm, genetic-based machine learning and classifier system. In Section III, we describe our adaptive architecture and the process of the architecture's self-organization, including principles and algorithms of width and depth extension. Section IV introduces the experiments and the results, together with the discussion and analysis. Finally, in Section V, a summarization of current architecture and a preview of future work are given.

2 Genetic Algorithm, Genetic Based Machine Learning and Classifier System

Genetic algorithms intent to get optimum solutions of a given problem by mechanics of natural selection and natural genetics Goldberg[6]. Genetic based machine learning (GBML) uses genetic algorithm to find and recombine new rules based on the hypothesize that new and better rules may be created by recombination of old ones Goldberg[6]. Classifier system is a rule-based learning system proposed by Holland. Being a common GBML architecture, classifier system adjusts the strength of each classifier from environmental feedback and discovers new rules using genetic algorithms Goldberg[6].

A classifier system consists of three sub-systems: rules and message system, credit assignment system and rule discovery system (see Fig.1).

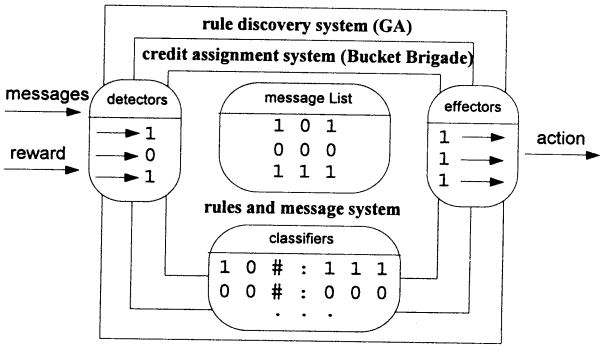


Fig.1 : Schematic of Classifier System

An environmental message, which is recognized by detectors, is sent to the rules and message system, where it is matched with the condition part of each classifier (normalized condition-action rule). The action part of those matched classifiers will be sent to effectors where corresponding action will be carried out. With the Bucket Brigade algorithm, the credit assignment system evaluates classifiers according to their relative usefulness to the system, i.e., their ability to make the system respond correctly to the environmental messages. In the rule discovery system, those useful rules will be utilized as "building blocks" to generate new and plausibly better rules under the operations of a genetic algorithm [Goldberg[6]].

In order to increase the adaptability of classifier systems under dynamic environment, some architectures have been proposed. M.Dorigo et al[5] developed an architecture of cognition based on both ethnologic and evolutionary considerations. Their work shows that hierarchical and parallel model is a plausible and powerful way to develop adaptive intelligent systems.

3 Self-organization Classifier System Tree

The autonomous agent must have the ability to adapt its cognitive structure to the dynamic environment. In order to do this, we proposed an adaptive architecture which can modify its structure in dynamical while interacting with the environment. In this section, we would like to give an overview of our adaptive classifier system tree. A complete model will be given in subsection A. The dynamic structure of the architecture and the principle of the structure's self-organization as well as two key mechanisms – width extension and depth extension will be described with detail in subsection B.

A The Complete Model

The basic unit of the architecture is referred as node that consists of a classifier system and a control unit. The control unit can feel the stimulation of the environment and make decision; the classifier system is the core of the cognition (Fig.2 [a]). There are many nodes working in parallel in the system. Each node learns a simple knowledge through interacting with the environment or other node. The goal of the whole system is to implement cognition through the coordination of the simple knowledge. The whole architecture is a tree-like one that has the property of fractal and self-similarity(Fig.2 [b]).

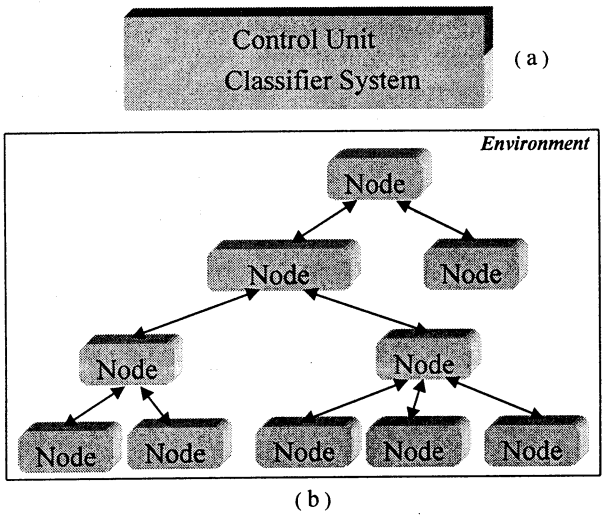


Fig. 2 : The Complete Model

We now give definitions of different knowledge.

Def 1 (Behavioral Knowledge) : Knowledge is called behavioral one if and only if the input message of the knowledge is directly from the environment.

Def 2 (Coordination Knowledge) : Knowledge is called coordination one *if and only if the input message of the knowledge is from other nodes, not the environment.*

With above definitions, we can continue our discussion. Typically, only leaf nodes response the stimulation of environment, so learning how to response behavioral knowledge is the main duty of a leaf node. By contrast, middle level nodes and root node have the responsibility to coordinate the behavior produced by lower level nodes, so learning coordination knowledge is the main duty of a middle level node and the root node. Specially, root node plays the most important role in the architecture, for all the behaviors will be coordinated by root node ultimately. And the total model is self-organized by the root node through inspecting the change of the environment as follows.

B Principle of Self-organization

The dynamic structure of the architecture and the principle of the structure's self-organization are show

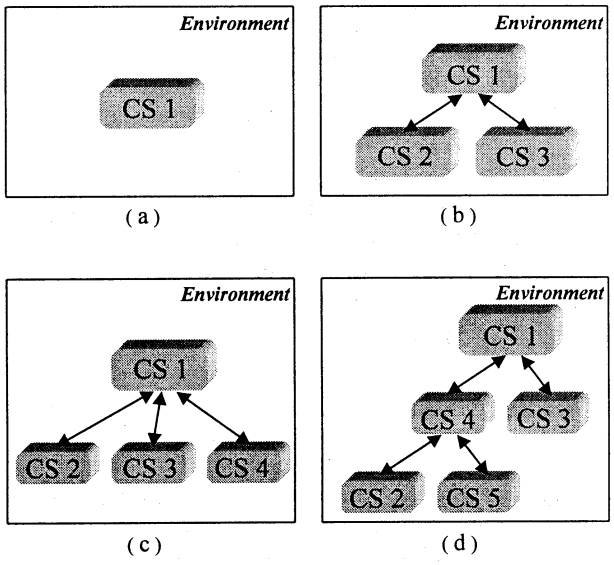


Fig.3 : Self Organization of the Architecture

in (Fig. 3.), where node is referred as 'CS'. We now describe the principle:

1) At the beginning (a), there is just one node — root node(CS1) in the system. The node can realize the stimulation of environment by its control unit and learn how to response it by its classifier system.

2) When environment changed and a new stimulation created (b), the root node can inspect the change. It will then create two new leaf nodes (CS2 and CS3) to respond the stimulation separately under the coordination of itself.

3) When environment changed again and a new stimulation created once more (c, d), the root node can also inspect the change. According to system's current status and the new stimulation's type, system will do ***width extension*** or ***depth extension***. Width extension (c) is that an appropriate parent node creates a new leaf node (CS4) directly to respond the stimulation. Depth extension (d) is that the system creates a middle level node (CS4) and assign it to create two leaf node (CS2 and CS5) to deal with two sub-type stimulation belong to the same type. After extension, this parent node (CS4) coordinates child nodes (CS2 and CS5) and transfers coordinating result to its upper levels until to the root node, which will decide the system's behavior ultimately.

4) Nodes related to certain stimulation will be deleted by the system under some special conditions so that the infinite increment of the tree can be avoided.

5) Whenever the root node inspects the change of the environment, the structure of the system will adapt to the environment by principles mentioned in 3) and 4).

We now summarize the main features of our architecture as follows:

1) The architecture is a tree-like one that has the property of fractal and self-similarity;

2) All the nodes in the system work in parallel;

3) Leaf nodes learn behavioral knowledge; middle nodes and root learn coordination knowledge respectively.

4) Once the root node inspects the change of the environment, it will drive the whole system reconstruct its architecture dynamically.

5) Once leaf nodes realize the stimulation of the environment, they will learn by their classifier systems to create the related behaviors and transfer them to their parent nodes.

6) Parent nodes in different levels will also learn by their classifier systems to coordinate the behaviors that are passed by their child nodes. The coordinating result will be passed to their upper level until to the root node, which will decide the system's behavior ultimately.

4 Experiments and Discussions

In this section, we would like give the experiments and discussions about our tree-like architecture. Our purpose in the experiment is to make sure the width extension and depth extension can be achieved by the agent itself under various environment settings. Firstly, in subsection A, we describe the simulation experiment settings. And then, in subsection B, we will discussion the details of using our model to learn a serial increasingly defined problems.

A Experiment Settings

Wilson[8] has proposed a simplified version of Holland's original classifier system. It is called Zeroth Level Classifier System (ZCS). Inspired by the facts that ZCS has been successfully used to deal with the animat problem [Wilson[9]], we use a ZCS as the cognition core in our adaptive architecture. Thus the node in our experiment can be illustrated as Fig. 4.

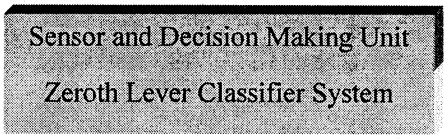


Fig.4 : A node in current experiment

M.Dorigo et al[5] proposed an experiment in which a simulated robot learns to follow a light source and meanwhile avoid a heat source. Our experiment is excited by what M.Dorigo did.

The experiment is about an animat following its food and avoiding its natural enemy. The settings can be described increasingly as follows:

Problem 1 (Simple Following) : In this problem, the animat intends to follow its food, which may be moving in a certain orbit.

Problem 2 (Following and Avoiding) : In this problem, the animat intends to follow the moving food, at the same time it should try its best to avoid a moving natural enemy.

Problem 3 (Following and Avoiding Two Things) : In this problem, the animat intends to follow the moving food, at the same time it should try its best to avoid two different moving natural enemies.

The increasing defined experiment settings will enable us to test whether our adaptive architecture can modify its structure in the fly. For either width extension or depth extension is needed in the serial of problems. We then discuss experiment result.

B Results and Discussions

We firstly put the simulated autonomous agent in the environment and let it to learn problem 1. Since there is just one behavior — following food in problem one, only one node — root node needed for achieve the cognition. It is shown that the root node can master the ability in a short time.

The second step is to increase the difficulty of learning. In order to do this, we add a natural enemy in the environment and the problem is changed to problem 2. Since the input message of food and the natural enemy are different. The root node can inspects it and do width extension (recall Fig.3[b]) to adapt the change of the environment. It is shown that the autonomous agent with the parallel cognition structure can master the ability of avoiding while following quicker than a agent do not possess it.

The third step is to increase the difficulty of learning again and make it even more complex. In order to do this, we add another natural enemy in the environment and thus there are two natural enemies the agent should avoid while following its food. The environment setting is thus changed to problem 3. Since the input message of food and the natural enemies are different. The root node will inspects the change and do depth extension (recall Fig.3[d]) to adapt the environment. It is shown that the mechanism of using different simple cognition units to response different kinds of knowledge (following or avoiding) and then coordinating the result with another unit is a better way to achieve complex cognition.

5 Conclusions

In this paper, an adaptive architecture has been proposed for extending genetic based machine learning in dynamic environment. The architecture has the properties of self-similarity and self-organization. There are two key mechanisms when the architecture organizes its structure on fly. One is width extension, which is used in case the new input message is exclusive. The other is depth extension, which is used in case the new input message is additive one. The experiment result shows that our self-organization architecture can achieve cognition in a high efficiency, it is due to the division of input message and the parallel running of nodes.

There are several things worth considering about. First, how to determine the kind of an input message. If message types are predefined, the advantage of self-organization architecture will be limited to a large extent. If all the message types are new to such an architecture, there is a problem when using the architecture to a real robot for its sensor must be made before running. There may be a trade-off between the architecture of cognition and the sensibility to environment in a real robot.

Second, although how to store and make use of experience knowledge has been introduced when doing width extension and depth extension, it is not enough for a real robot. When constructing a real robot, or in the future, constructing an artificial life body, its cognition architecture must have the ability to do planning, scheduling and to make decisions as well as respond to the environment. We believe nature is the exhaustless source to borrow from. Under coordination of evolution algorithms, a hybrid architecture including expert system, neural network, petri net as well as our adaptive classifier system may be a plausible way to achieve this.

Third, our model and experiment so far just touch upon one autonomous agent. In the real world, the coordination of multi-agents will be a very important research domain. Whether our self-organization architecture can be extend and then used in this domain? It is a problem worth thinking.

References

1. Buchanan, Bruce G. *Rule-based expert systems : the MYCIN experiments of the Stanford Heuristic Programming Project*, Addison-Wesley, 1984.

2. J.H.Holland and J. S. Reitman. Cognitive systems based on adaptive algorithms. *Pattern directed inference systems*, pages 313-329, Academic Press, New York, 1978.

3. J.H.Holland.Escaping brittleness: the possibilities of general-purpose learning algorithms applied to parallel rule-based systems. *Machine Learning, an artificial intelligence approach*, 2. Morgan Kaufmann, Los Altos, California, 1986.

4. R. A. Brooks, A robust layered control system for a mobile robot, *Artificial Intell. Memo 864*, MIT, Cambridge, MA, 1985.

5. Marco Dorigo et al. Genetics-based maching learning and behavior-based robotics: a new synthesis. *IEEE Transactions on systems, man and cybernetics*, 23(1):141-153, Jan. 1993.

6. D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, Jan. 1989.

7. Hee-Heon Song and Seong-Whan Lee, A Self-Organizing Neural Tree for Large-Set Pattern Classification, *IEEE Transactions on Neural Networks*, Vol. 9, No. 3, May 1998

8. S.W.Wilson. ZCS: a zeroth level classifier system. *Evolutionary Computation*, 2(1):1-18, 1994.

9. S. W. Wilson. Classifier systems and the Animat problem. *Machine Learning*, 2(3):199-228, 1987.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

WINNING STRATEGIES FOR ROBOT WAR GAMES

Boris Stilman
Department of Computer Science and Engineering, University of Colorado at Denver
Campus Box 109, Denver, CO 80217-3364, USA
Email: bstilman@cse.cudenver.edu

Abstract: Linguistic Geometry (LG) includes the syntactic tools for *reasoning* about multiagent concurrent pursuit-evasion games. This approach based on mathematical modeling of expert heuristics. It gives us powerful tools for reducing the search space in various problems by introducing a concurrent game and decomposing it into a hierarchy of dynamic interacting subsystems. This paper introduces a reader to a series of search problems (robot war games) of gradually increased dimension where the LG tools have been successfully applied. We may suggest that these tools allowed us to distinguish a new class of low complexity problems among those that are usually considered as NP-complete.

Keywords: games, simulation, linguistic geometry, concurrent system, search heuristics, multiagent system.

1. Computational Challenges

Land combat operations, aircraft combat missions, tracking and possible interception of missiles and satellites, surveillance operations, etc. can be unified as problems of optimal operation of concurrent multiagent systems. Unmanned aircraft or tanks may participate in reconnaissance missions or in the full scale combat operation. Similar teams of intelligent vehicles may be dispatched by the adversary. Control of those activities requires permanent adaptation to the intermediate results and dynamic re-computation in real time. Space combat simulation problems are, in general, similar to the other military combat simulations. However, the astrodynamics of the spacecraft makes these problems significantly more complex. Another factor is the vehicle's autonomy. While the autonomy of the land, navy, and aerial vehicles is highly desirable, it is essential for the spacecraft, especially, if they are away from Earth. Simulation and control of the two-three spacecraft combat requires enormous amounts of computations. Different multiagent systems like problems of intelligent manufacturing, software re-engineering, network integrity (Internet Cyberwar), etc. can be represented as combat simulations. Conventional models of combats can achieve computational and, sometimes, analytical solutions for the simple cases. Real world cases employing those approaches are, however, computationally intractable.

None of the conventional approaches to the above problems allows to scale up to the real world systems with respect to the number of agents, dynamic change of their capabilities, size (and dimension) of the operational district, concurrent actions, real time requirements, etc. One of the main difficulties is the enormous complexity of computations due to the exponential growth of the number of variants of the system's operation to be analyzed.

2. Development of Linguistic Geometry

Linguistic Geometry (LG) (Stilman, 1993-1998) includes the syntactic tools for *knowledge representation* and *reasoning* about concurrent multiagent systems by modeling them as discrete pursuit-evasion games. The LG tools provide a framework for the evaluation of computational complexity and accuracy of solutions, for generating computer programs for specific problem domains. LG allowed us to discover the inner properties of human expert heuristics that are successful in a certain class of games. This approach provides us with an opportunity to transfer formal properties and constructions from one problem to another and to reuse tools in a new problem domain. In a sense, it is the application of the method of a chess expert to robot control or maintenance scheduling and vice versa.

What do we know about the method of a chess expert? Of course, a computer is the perfect tool for discovering and modeling such a method. The history of computer chess began with a paper by Professor Claude Shannon (1950) in which he introduced the framework that guided further development. Employing mostly the brute force search, relying ultimately on computer speed, computer chess programs gradually increased their level of playing (Newborn, 1996). In the middle of the 90s this level reached the one of a grandmaster. After the May 1997 historical event, when the Deep Blue computer chess system defeated World Chess Champion Gary Kasparov, this problem lost its exciting attractiveness. In the June 6, 1997 issue of *Science* magazine Professor John McCarthy (1997) wrote: "In 1965 the Russian mathematician Alexander Kronrod said, "Chess is the Drosophila of Artificial Intelligence." However, computer chess has developed much as genetics might have if the geneticists had concentrated their efforts starting in 1910 on breeding racing Drosophila. We would have some science, but mainly we would have very fast fruit flies."

All the major advances in computer chess, including the Deep Blue system triumph, were related to the brute force approach. What can we learn from these advances for

different problems, particularly, for the problems of much higher dimension? Not much. Even in the future we would not be able to solve these problems employing the brute force. The grandmaster's approach (of almost no search) has not been discovered yet. After the 1997 event, it is more important than ever before, that chess stays as a scientific Drosophila of AI and not just a racer (McCarthy, 1990, 1997).

Not all the research in computer chess went in the direction of the brute force. In the 70s and 80s project PIONEER led by the Former World Chess Champion, Professor Mikhail Botvinnik was developed in Moscow, Russia. In his book (1984) Botvinnik writes that the brute force method is "hardly capable of further progress. It is the computer turn to adopt a more fruitful method - perhaps PIONEER. And if PIONEER is unsuccessful, we must believe that other method will be found. The problem must and will be solved."

LG is the direct successor of the project PIONEER. Since 1980, the theoretical foundations of the models constructed in the course of this project have been developed. In 1991, this approach was named Linguistic Geometry (LG). A reason for this name is the geometrical nature of discovered heuristics and the mathematical tools of the theory of formal languages used to formalize them. A story behind this formalization is as follows. In the 1960's, a formal syntactic approach to the investigation of properties of natural language resulted in the fast development of a theory of formal languages by Chomsky (1963), Ginsburg (1966), and others. This development provided an interesting opportunity for dissemination of this approach to different areas. In particular, there came an idea of analogous linguistic representation of images. This idea was successfully developed into syntactic methods of pattern recognition by Pavlidis (1977), Fu (1974), and picture description languages by Shaw (1969), and others. Searching for the adequate mathematical tools formalizing human heuristics of dynamic hierarchies, the author transformed the idea of linguistic representation of complex real world and artificial images into the idea of similar representation of complex hierarchical systems (Stilman, 1985). However, the appropriate languages possess more sophisticated attributes than languages usually used for pattern description. The origin of such languages can be traced back to the research on programmed attribute grammars by Knuth (1968), Rozenkrantz (1969), and others. A mathematical environment for the formal representation of this model based on the first order predicate calculus was developed following the theories of formal problem solving and planning by McCarthy and Hayes (1969), Nilsson (1980), and others.

3. Complex Systems

The LG model must interpret the definition of Complex System. Formally, *Complex System* is the following eight-tuple (Stilman, 1993a):

$$< X, P, R_p, \{ON\}, v, S_i, S_t, TR>, \text{ where}$$

— $X=\{x_i\}$ is a finite set of *points*; locations of elements;

— $P=\{p_i\}$ is a finite set of *elements*; P is a union of two non-intersecting subsets P_1 and P_2;

— $R_p(x, y)$ is a set of binary relations of *reachability* in X (x and y are from X, p from P);

— ON(p)=x, where ON is a partial function of *placement* from P into X;

— v is a function on P with positive integer values describing the *values* of elements.
The Complex System searches the state space, which should have initial and target states;

— S_i and S_t are the descriptions of the *initial* and *target* states in the language of the first order predicate calculus, which matches with each relation a certain formula. Thus, each state from S_i or S_t is described by a certain set of formulas $\{ON(p_j) = x_k\}$;

— TR is a set of operators, TRANSITION(p, x, y), of transitions of the System from one state to another one. These operators describe the transition in terms of two lists of formulas (to be removed from and added to the description of the state), and a list of applicability of the transition. Here,

Remove list: ON(p) = x, ON(q) = y;
Add list: ON(p) = y;
Applicability list: $(ON(p) = x) \wedge R_p(x,y)$,

where p belongs to P_1 and q belongs to P_2 or vice versa. The transitions are carried out with participation of a number of elements p from P_1, P_2.

The elements of the Complex System are divided into two subsets P_1 and P_2. They might be considered as units moving along the reachable points (locations). Element p can move from point x to point y if these points are reachable, i.e., $R_p(x, y)$ holds. The current location of each element is described by the equation ON(p) = x. Thus, the description of each state of the System $\{ON(p_j) = x_k\}$ is the set of descriptions of the locations of elements. The operator TRANSITION(p, x, y) describes the change of the state of the System caused by the move of the element p from point x to point y. The element q from point y must be withdrawn (eliminated) if p and q do not belong to the same subset (P_1 or P_2). The problem of the optimal operation of the System is considered as a search for the optimal sequence of transitions leading from the initial state of S_i to a target state of S_t.

To avoid exhaustive search in Complex Systems we introduce new representation: a hierarchy of formal languages (Stilman, 1993a, 1993b, 1993c, 1997c, 1998c).

4. Series of Combat Simulations

In Sections 5-7 we consider a series of examples of combat simulation problems. These are Reti-like problems, i.e.,

the generalizations of the R.Reti endgame (1920), (Fig. 1). They are simple enough to be used as the first demonstration of the LG approach. On the other hand, they are not trivial and require significant search to be solved employing conventional approaches. Professor McCarthy (1998) writes: "Note that Reti's idea can be implemented on a 100×100 board, and humans will still solve the problem, but present (conventional, i.e. brute force - B.S.) programs will not .... Chess can serve as a Drosofila for AI if AI researchers try to make a program that (will) come up with the idea needed to solve the problem on a board of arbitrary size."

A series of three problems includes generalization suggested by Professor McCarthy. The first problem is just a reformulation of the R. Reti endgame. The second is a similar problem in the 3D space/board, while the third is a totally concurrent version of the Reti endgame for the n×n board. All three of them and many different, significantly more complex problems have been solved employing LG tools (Stilman, 1998c). This result is exciting because the solutions are constructive: they have been obtained without search and with simultaneous proof of optimality. In particular, it can be suggested that these problems are representatives of a wider class of problems of low computational complexity (probably, $O(n^4)$). This would be a new subclass in a class of NP-complete problems. It is likely that many real world problems considered in Section 1 including robot combat simulations are members of this subclass.

5. 2D/4A Robot Combat

Consider a problem of simplified air combat with 4 aircraft and 2D operational district. Basically, we have to define an operational district, the set of *points* X, mobile units, i.e., the set of *elements* P broken in two subsets-opposing sides, P_1 and P_2, and moving abilities of the elements, *relations of reachability* $R_p(x, y)$.

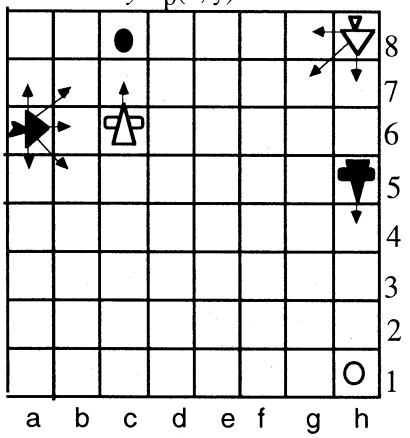


Figure 1. 2D serial model for 8×8 district.

Robots with various moving abilities are shown in Fig. 1. The operational district X is represented as 2D 8×8 square grid. Robot W-FIGHTER (White Fighter) standing on h8, can move to any next square (shown by arrows). The other robot B-BOMBER (Black Bomber) from h5 can move only straight ahead, one square at a time, e.g., from h5 to h4, from h4 to h3, etc. Robot B-FIGHTER (Black Fighter) standing on a6, can move to any next square similarly to robot W-FIGHTER (shown by arrows). Robot W-BOMBER standing on c6 is analogous with the robot B-BOMBER; it can move only straight ahead but in reverse direction. Thus, robot W-FIGHTER on h8 can reach any of the points $y \in \{h7, g7, g8\}$ in on step, i.e., $R_{W\text{-}FIGHTER}(h8, y)$ holds, while W-BOMBER can reach only c7 in one step.

Assume that robots W-FIGHTER and W-BOMBER belong to one side, while B-FIGHTER and B-BOMBER belong to the opposing side: W-FIGHTER $\in P_1$, W-BOMBER $\in P_1$, B-FIGHTER $\in P_2$, B-BOMBER $\in P_2$. Also assume that two more robots, W-TARGET and B-TARGET, (unmoving devices or target areas) stand on h1 and c8, respectively. W-TARGET belongs to P_1, while B-TARGET $\in P_2$. Each of the BOMBERs can destroy unmoving TARGET ahead of the course; it also has powerful weapons able to destroy opposing FIGHTERs on the next diagonal squares ahead of the course. For example W-BOMBER from c6 can destroy opposing FIGHTERs on b7 and d7. Each of the FIGHTERs is capable to destroy an opposing BOMBER approaching its location, but it is also able to protect its friendly BOMBER approaching its prospective location. In the latter case the joint protective power of the combined weapons of the friendly BOMBER and FIGHTER can protect the BOMBER from interception. For example, W-FIGHTER located at d6 can protect W-BOMBER on c6 and c7.

The combat considered can be broken into two local operations. The first operation is as follows: robot B-BOMBER should reach point h1 to destroy the W-TARGET, while W-FIGHTER will try to intercept this motion. The second operation is similar: robot W-BOMBER should reach point c8 to destroy the B-TARGET, while B-FIGHTER will try to intercept this motion. After destroying the opposing TARGET the attacking side is considered a winner of the local operation and the global battle. The only chance for the opposing side to avenge is to hit its TARGET on the next time increment and this way end the battle in a draw. The conditions considered above give us S_t, the description of target states of the Complex System. The description of the initial state S_i is obvious and follows from Fig. 1.

Assume that motions of the opposing sides *alternate* and due to the shortage of resources (which is typical in a real combat operation) or some other reasons, each side *can not* participate in both operations simultaneously. It means that during the current time increment, in case of White turn, either W-BOMBER or W-FIGHTER can move. Analogous condition holds for Black. Of course, it does not mean that if one side began participating in one of the

operations it must complete it. Any time on its turn each side can switch from one operation to another, e.g., by transferring resources (fuel, weapons, human resources, etc.), and later switch back. An example without these restrictions is considered in Section 7.

It seems that local operations are independent, because they are located far from each other. Moreover, the operation of B-BOMBER from h5 looks like unconditionally winning operation, and, consequently, the global battle can be easily won by the Black side.

Is there a strategy for White to make a draw?

The specific formal question is as follows. Is there an optimal strategy that provides one of the following:

1. Both BOMBERs hit their targets and stay safe for at least one time increment, or
2. Both BOMBERs are destroyed before they hit their targets or immediately after that?

Of course, it can be answered by the direct search employing, for example, minimax algorithm with alpha-beta pruning. Experiments with computer programs and theoretical evaluations showed that in order to solve this problem employing conventional approaches the search tree must include at least $\sqrt{(9^{13})} \sim 1.5$ million moves.

A solution generated by the LG tools without search is shown in Fig. 2.

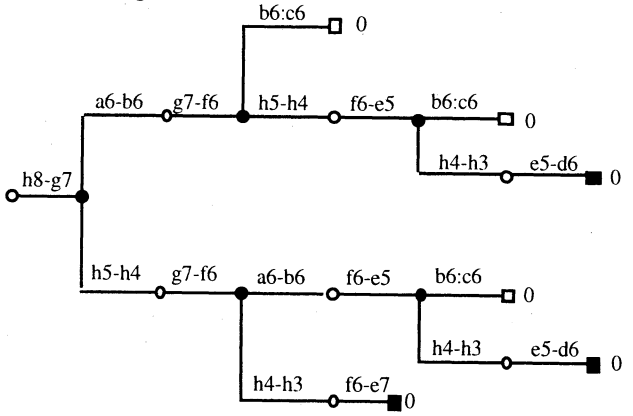


Figure 2. Solution to 2D/4A problem.

This solution can be explained as follows. At first, the W-FIGHTER should move along the diagonal of the square district h8-g7-f6, etc. The following strategy depends on the Black actions. If B-BOMBER is active and moving to the destination h1 then W-FIGHTER will have enough time to approach W-BOMBER and protect it. Otherwise, if B-FIGHTER is active and moving to intercept W-BOMBER then W-FIGHTER will have enough time to intercept B-BOMBER before it hits the TARGET or immediately after that. The number of moves "searched" is just the number of moves included in the solution (Fig. 2). The construction and the proof of optimality are presented in (Stilman, 1998a, 1998b, 1998c).

6. 3D/4A Robot Combat

Consider a 3D version of the 2D/4A problem. Space robot vehicles with various moving abilities are shown in Fig. 3. The operational district X is the 3D grid of 8×8×8. Robot W-INTERCEPTOR (White Interceptor) located at 118 (x = 1, y = 1, z = 8), can move to any next location, i.e., 117, 217, 218, 228, 227, 128, 127. The other robotic vehicle B-STATION (double-ring shape in Fig. 3) from 416 can move only straight ahead towards the goal area 816 (shaded in Fig. 3), one square at a time, e.g., from 416 to 516, from 516 to 616, etc. Robot B-INTERCEPTOR (Black Interceptor) located at 186, can move to any next square similarly to robot W-INTERCEPTOR. Robotic vehicle W-STATION located at 266 is analogous with the robotic B-STATION; it can move only straight ahead towards the goal area 268 (shaded in Fig. 3). Thus, robot W-INTERCEPTOR on 118 can reach any of the points y $\in$ {117, 217, 218, 228, 227, 128, 127} in one step, i.e., $R_{W\text{-}INTERCEPTOR}(118, y)$ holds, while W-STATION can reach only 267 in one step.

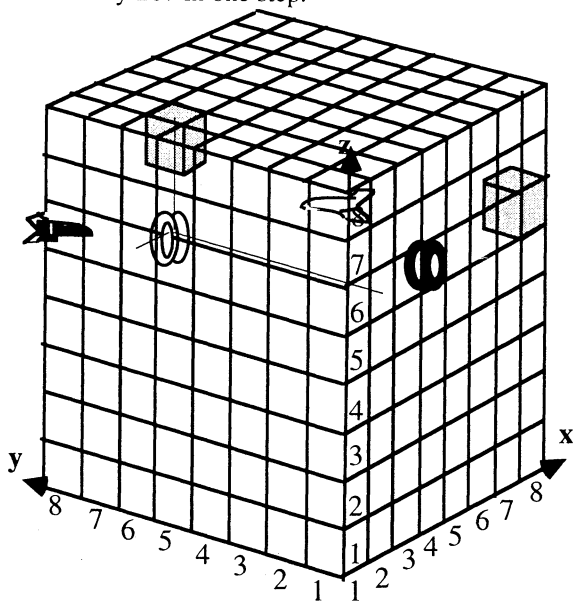


Figure 3. 3D/4A space robot combat.

Assume that robots W-INTERCEPTOR and W-STATION belong to one side, while B-INTERCEPTOR and B-STATION belong to the opposite side: W-INTERCEPTOR $\in P_1$, W-STATION $\in P_1$, B-INTERCEPTOR $\in P_2$, B-STATION $\in P_2$. Also assume that both goal areas, 816 and 268, are the safe areas for B-STATION and W-STATION, respectively, if station reached the area and stayed there for more than one time interval. Each of the STATIONs has powerful weapons capable to destroy opposing INTERCEPTORs at the next diagonal locations ahead of the course. For example W-STATION from 266 can destroy opposing INTERCEPTORs at 157, 257, 357, 367, 377, 277, 177, 167. Each of the INTERCEPTORs is capable to destroy an opposing STATION approaching its location from any direction, but it also capable to protect its friendly STATION approaching its prospective location. In the latter case the joint protective power of the combined weapons of the friendly STATION and INTERCEPTOR

(from any next to the STATION area) can protect the STATION from interception. For example, W-INTERCEPTOR located at 156 can protect W-STATION on 266 and 267.

As in the 2D case, the combat can be broken into two local operations. The first operation is as follows: robot B-STATION should reach the strategic point 816 safely and stay there for at list one time interval, while W-INTERCEPTOR will try to intercept this motion. The second operation is similar: robot W-STATION should reach point 268, while B-INTERCEPTOR will try to intercept this motion. After reaching the designated strategic area the (attacking) side is considered as a winner of the local operation and the global battle. The only chance for the opposing side to avenge is to reach its own strategic area within the next time interval and this way end the battle in a draw. The conditions considered above give us S_t, the description of target states of the Complex System. The description of the initial state S_i is obvious and follows from Fig. 3. Analogously with 2D/3A problem only one vehicle at a time can move and Black and While alternate turns. It is White turn at the Start State. Is there a strategy for the White to make a draw?

A solution to this problem is similar with 2D/4A problem. Of course, it can be found by the direct search employing, for example, minimax algorithm with alpha-beta pruning. Theoretical evaluation and experiments with the computer programs showed that the search would require billions of moves, at least $\sqrt{(27^{13})}$. The solution obtained by the LG tools by construction of strategies is as follows. At first, the W-INTERCEPTOR should move along the main diagonal of the 3D district 118-227-336, etc.. It must not follow the main diagonal exactly; minor deviations are possible. The following strategy depends on the Black actions. If B-STATION is active and moving to destination then W-INTERCEPTOR will have enough time to approach W-STATION and protect it. If B-INTERCEPTOR was active and moved to intercept W-STATION then W-INTERCEPTOR would have enough time to intercept B-STATION. The construction and the proof of optimality are similar to the 2D case (Stilman, 1998a, 1998b, 1998c). The preliminary results are published in (Stilman, 1996a, 1998c).

7. Totally Concurrent Combat

Consider another generalization of the 2D/4A problem. Robots with various moving abilities are shown in Fig. 4. The operational district X is the n×n grid, n>7. Robots W-FIGHTER, W-BOMBER, B-FIGHTER and B-BOMBER and their moving abilities are similar with those from the 2D/4A problem (Section 5). The main difference is that all the units can move *simultaneously*. It means, for example, that during the current time increment, all four vehicles, W-BOMBER, W-FIGHTER, B-BOMBER, and B-FIGHTER, three of them, two, one, or none of them can move.

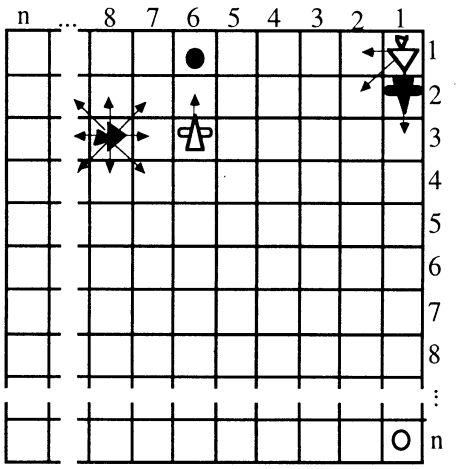


Figure 4. 2D totally concurrent model for the n×n district.

Other differences are as follows. Each of the FIGHTERs is able to destroy an opposing BOMBER by moving to its location, but it also able to destroy an opposing BOMBER if this BOMBER itself arrives at the current FIGHTER's location. For example, if the B-FIGHTER is at location (6, 1) and W-BOMBER arrives there (unprotected) then during the same time increment it destroys the B-TARGET and is destroyed itself by B-FIGHTER. Each BOMBER can be protected by its friendly FIGHTER analogously with 2D/4A problem.

Each of the BOMBERs is vulnerable not only to a FIGHTER's attack but also to the explosion of another BOMBER. If W-FIGHTER hits B-BOMBER while the latter is fully armed, i.e., it is not at its final destination – square (1, n), and W-BOMBER is moving during the same time increment, it will be destroyed as a result of the B-BOMBER's explosion. If W-BOMBER is not moving at this moment it is safe. Similar condition holds for the B-BOMBER: it can not move at the moment when W-BOMBER is being destroyed (excluding (6, 1) where destruction might happen only after the W-BOMBER hits the B-TARGET).

The combat considered can be broken into two local operations. The first operation is as follows: robot B-BOMBER should reach location (1, n) to destroy the W-TARGET, while W-FIGHTER will try to intercept this motion. The second operation is similar: robot W-BOMBER should reach location (6, 1) to destroy the B-TARGET, while B-FIGHTER will try to intercept this motion. After destroying the opposing TARGET and keeping its BOMBER safe (for at least one time increment), the attacking side is considered a winner of the local operation and the global combat. The only chance for the opposing side to avenge is to hit its TARGET and this way end the battle in a *draw*. The question to be answered is as follows: *Is there a strategy for White to make a draw?* In other words: Is there an optimal strategy that provides one of the following:

1. Both BOMBERs hit their targets and none of the BOMBERs is destroyed at the moment of strike, or
2. Both BOMBERs are destroyed before they hit their targets or at the moment of strike?

Exhaustive search algorithm would have to generate a 324^{n-1} search tree. Application of alpha-beta pruning to the concurrent games requires additional research. However, with this size of the full search tree we do not have much hope for a significant search reduction.

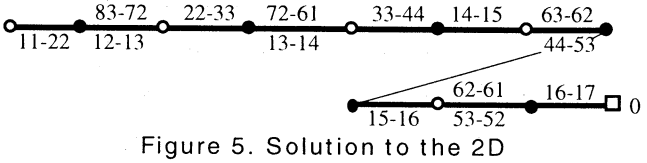


Figure 5. Solution to the 2D concurrent problem.

A solution to this problem, obtained employing the LG tools *without search*, is shown in Fig. 5. Every concurrent move is represented by *two consecutive arcs*. The arc outgoing the white node represents the White component of a concurrent move, the concurrent motions of the White side, while the arc outgoing the black node represents the Black component of the same move. The key point of the solution is that W-BOMBER must wait at (6,3) until W-FIGHTER approaches it and then move to the B-TARGET under protection. If W-BOMBER is destroyed before the protector arrives, the W-FIGHTER will have enough time to intercept the B-BOMBER. Construction of the solution and optimality proof are similar with 2D/4A case (Stilman, 1998a, 1998b, 1998c). Preliminary results are considered in (Stilman, 1998c), (Stilman, Fletcher, 1998).

References

Botvinnik, M.M. (1984). *Computers in Chess: Solving Inexact Search Problems.* Springer Series in Symbolic Computation, New York: Springer-Verlag, translation from Russian edition by Sovetskoe Radio, 1978.

Chomsky, N. (1963). Formal Properties of Grammars. in *Handbook of Mathematical Psychology*, eds. R.Luce, R.Bush, E. Galanter., vol. 2 (323-418). New York: Wiley.

Fikes, R.E. and Nilsson, N.J. (1971). STRIPS: A New Approach to the Application of Theorem Proving in Problem Solving, *Artificial Intelligence* 2 (189–208).

Fu, K.S. (1974). *Syntactic Methods in Pattern Recognition*, Acad. Press, New York.

Ginsburg, S. (1966). *The Mathematical Theory of Context-Free Languages*, McGraw Hill, New York.

Knuth, D.E. (1968) Semantics of Context-Free Languages, *Mathematical Systems Theory*, (127–146), 2.

McCarthy, J. (1990). Chess as the Drosophila of AI, in Computers, Chess, and Cognition, Ed. by T. Anthony Marsland, Jonathan Shaeffer, (227-237), Springer-Verlag, New York.

McCarthy, J. (1997). AI as Sport, Review of (Newborn, 1996), (1518-1519), *Science*, 276, June 6, 1997.

McCarthy, J. (1998). "Computer Chess" and human chess, http://www-formal.stanford.edu/jmc/reti.html

McCarthy, J. and Hayes, P.J. (1969). Some Philosophical Problems from the Standpoint of Artificial Intelligence. *Machine Intelligence* (463–502), 4.

Newborn, M. (1996). *Computer Chess Comes of Age*, Springer-Verlag, New York, NY.

Nilsson, N.J. (1980). *Principles of Artificial Intelligence*, Palo Alto, CA: Tioga Publ.

Pavlidis, T. (1977). *Structural Pattern Recognition*, New York: Springer-Verlag.

Rozenkrantz, D.J. (1969). Programmed Grammars and Classes of Formal Languages, *J. of ACM* (107–131), 1.

Shannon, C.E. (1950). Programming a digital computer for playing chess, *Philosophy Magazine*, March, (356-375), 41.

Shaw, A.C. (1969). A Formal Picture Description Scheme as a Basis for Picture Processing System, *Information and Control* (9-52), 19.

Stilman, B. (1985). Hierarchy of Formal Grammars for Solving Search Problems, *Proc. of the Int. Workshop* (63-72), MDNTP, Moscow, (in Russian).

Stilman, B. (1993a). A Linguistic Approach to Geometric Reasoning, *Int. J. Computers and Mathematics with Applications* (29-57), 26(7).

Stilman, B. (1993b). Network Languages for Complex Systems, *Int. J. Computers and Mathematics with Applications* (51-79), 26(8).

Stilman, B. (1993c). Syntactic Hierarchy for Robotic Systems, *Integrated Computer-Aided Engineering*, (57-81), 1(1).

Stilman, B. (1993d). A Formal Language for Hierarchical Systems Control, *Languages of Design* (333-356), 1(4).

Stilman, B. (1994a). Translations of Network Languages. *Int. J. Computers and Mathematics with Applications* (65-98), 27(2).

Stilman, B., (1994c) A Linguistic Geometry for Control Systems Design, *Int. J. of Computers and Their Applications*, (89-110), Vol. 1, No. 2, Dec. 1994.

Stilman, B., (1996) Network Languages for Intelligent Control, *Int. J. Computers & Mathematics with Applications*, (91-118), 31(3).

Stilman, B., (1997c). Managing Search Complexity in Linguistic Geometry, IEEE *Trans. on Systems, Man, and Cybernetics*, Vol. 27, No. 6, pp. 978-998, Dec.

Stilman, B., Fletcher, C., (1998) Systems Modeling in Linguistic Geometry: Natural and Artificial Conflicts, *Int. J. Systems Analysis, Modeling, Simulation*, pp. 1-42, May 1998.

Stilman, B. (1998a). No-Search Approach in Linguistic Geometry: State Space Chart, (4907-4912), *Proc. of the IEEE Int. Conf. on Systems, Man, and Cybernetics — SMC'98*, San Diego, CA, USA, Oct. 11-14, 1998.

Stilman, B. (1998b). No-Search Approach in Linguistic Geometry: Construction of Strategies, (4913-4918), *Proc. of the IEEE Int. Conf. on Systems, Man, and Cybernetics — SMC'98*, San Diego, CA, USA, Oct. 11-14, 1998.

Stilman, B., (1998c) *Linguistic Geometry: From Search to Construction*, 300 pp., Kluwer Acad. Publ., (to appear).

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Insect size robots with insect level intelligence

J.D. Nicoud

Swiss Federal Institute of Technology
Microprocessor and Interface Lab
LAMI-EPFL
CH-1015 Lausanne Switzerland
Tel ++41 21 693-2642 Fax ++41 21 693-5263
Email jean-daniel.nicoud@epfl.ch http://diwww.epfl.ch/lami/

Abstract

Miniaturization of robots is a general trend in order to facilitates the experiments of insect-like behaviours. The paper desribe the various problems one meet in this miniaturization process, with examples given for 1 to 6 centimeter robots.

Key words

Robots, miniaturization, autonomy

Introduction

Insects are simple creatures some robot engineers get their inspiration from. Building an insect-like robot, both in size and intelligence, will be the dream that will motivate many researchers deep inside the 21st Century. The major problem may not be miniaturization of parts and processing, but sensors, actuators and power sources.

The biggest insect is about 10 cm long. This correspond to a volume of 1 cubic inch. For several years now, the Nagoya microrobot contest proposes a simple race between two walls for an autonomous 1 inch cube robot, which can be tele-operated. Plenty of original solutions are proposed, but we are still far from a reliable behaviour.

The Nagoya contest proposes also a 1 cm3 category, with an external power source. A processor can be included once in this volume, but nobody has been able yet to add some sensor. These contests are for advanced students, frequently supported by the staff of an experienced and motivated lab. Progress are very slow. No partial solution is satisfactory. Nobody is talking about intelligent behaviours. Why?

Robot and insect architecture

It is not so easy to build a miniature robot. An autonomous system, being a microrobot or an insect, consists of the followings:
- A chassis (skeletton)
- A set of actuators (muscles)
- A low level controller (spinal chord)
- An autonomous navigation controller (brain)
- A battery (glucose)
- A recharging system (mandibules and stomach)

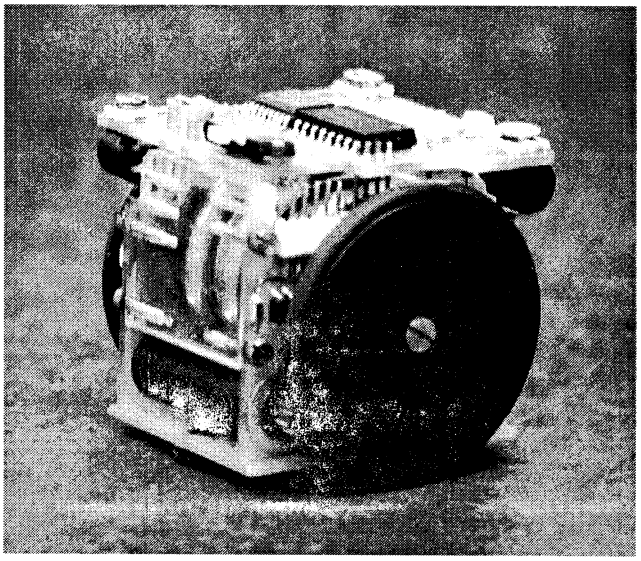
Let's go through all the associated implementation problems.

Chassis and wheels

Insects are legged. But they have muscles to activate them. These are impressive modular actuators, with a high force-to-weight ratio, good speed of response, high reliability, high efficiency, local energy storage, long stroke length, and proprioception as a feedback for the real elongation. Several muscles can be found into the 20 micron leg of a 1 mm insect and the same technology is used in elephants.

We believe engineers can ignore legged robots for insect-size implementations as long an actuator like a muscle is not invented. SMA, piezo devices and contractile polymer exists, but none allows for a contractions of more than a few percent, and not the required 50%. The best electric motors have a weight/power ratio similar to muscles, but they need reduction gears, motion and torque sensors and a huge power supply. Insect size autonomous robots will have wheels or stick and slip pads and stay on a flat surface. They may swim (float or sink), but I am afraid they will never fly.

The chassis of a micro robot is not the major problem, but one need to reduce the part numbers and the assembly cost. PC board is an excellent construction material, with the advantage to hide bunch of wires and increase the reliability of the connections. The Khepera robot [1] has proven that even in a 6cm diameter robot, no chassis is a reliable approach. The Inchy chassis (figure 1) is built from 2 PC boards maintained by two simple pieces of plexiglas. With small connectors, they link the two active circuits (processor and sensors). Another construction used with the Alice robot [2] is based on a flexible PC boards maintained in a frame (figure 2).



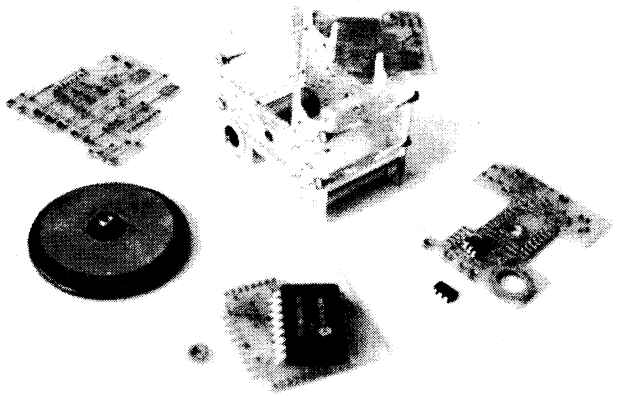
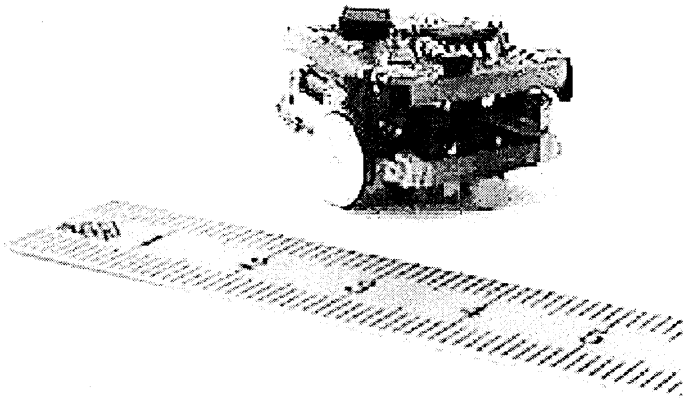


Fig 1. Inchy : A 1 cubic inch autonomous robot

Reduction gears, when required and not included in the motor, are integrated into the chassis. Like legs, they are part of the skeletton. They add reasonable constraints to the chassis design, however more difficult to satisy when the external size is limited, as for a 1 cc microrobot (figure 3).



The Nagoya contest is a good place to see the diversity of solutions for moving a microrobot. Due to the difficulty to get a small motor, and reduce its rotation speed, wobbles motors, stick and slip solutions, inertia motors have been demonstrated with some success on the 1cc tele-powered Nagoya contest robots. None of these low efficiency solutions were used on the autonomous 1 cubic inch robots.

Fig 2. Alice: A 2cm autonomous robot with ETA motors

For an autonomous robot, there are three basic solutions for the motion motors. DC motors are easy to implement, but the smallest ones are 8 mm in diameter (power 0.17 Watt). Smaller 6 mm pager motors exist, with an increased current and smaller lifetime. These DC motors need an encoder or some feedback to measure distence or be sure the robot heads toward the good direction.

Stepping or synchronous motors are easy to control. The smoovy motors from RMB [3] are 3 and 5 mm in diameter. Adequate software allows to do microsteps [4,5] and enter a power save mode when the motor is stopped. The 1.9 mm motor from Minimotor is quite impressive, but with its 0.5 Amps at 0.6V, it seems difficult do pack its complex control electronics in a small autonomous robot. All these motors spin at very high speed, but are available with an expensive planetary gear sporting reduction ratios of 20 to 100.

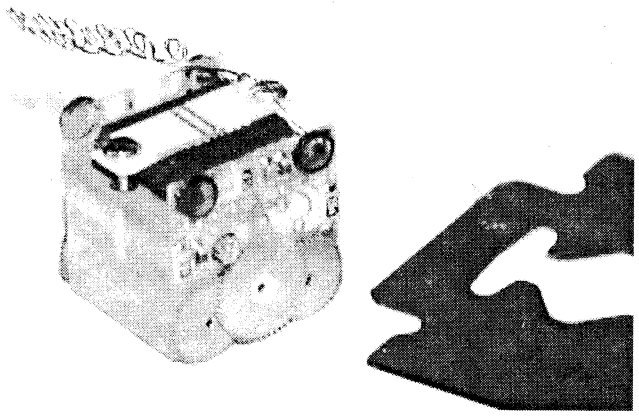


Fig 3. Jemmy : a 1cc robot with a processor

The third solution is to use watch "Lavet" motors. These motors are very low power (and low torque). Bidirectional implementation are available [6] and are controlled like a synchronous motor. The major advantage is these motors include reduction gears. The ETA motor used on the Alice weight 1 gram and uses less than 5mW of power. The coils are directly controlled by the outputs of a small microcontroller. The Munshu from Epson uses a specially designed pair of unidirectionnal Lavet motor and an ASIC, which leaves enough space for a battery and two photosensors. It is the smallest autonomous robot up to now.

Using the ETA motor for building a 1 cubic inch robot is reasonably easy now with the Microrobot kit [6] based on a PIC microprocessor. The larger Switec motor is easier to implement; the SwiBot (figure 4) is a nice low cost implementation using two Switec motors and a on site reprogrammable PIC 16F84 as main processor. It accepts standard or custom made extension boards with rather simple size and power supply constraints compared to one cubic inch robots. For those who are ready to program in assembler, the Swibot is a wonderfull educational tool and can be used for research too. The Khepera is of course a higher performance robot with a good set of options, for those who like to program in C the most sophisticated algorithms.

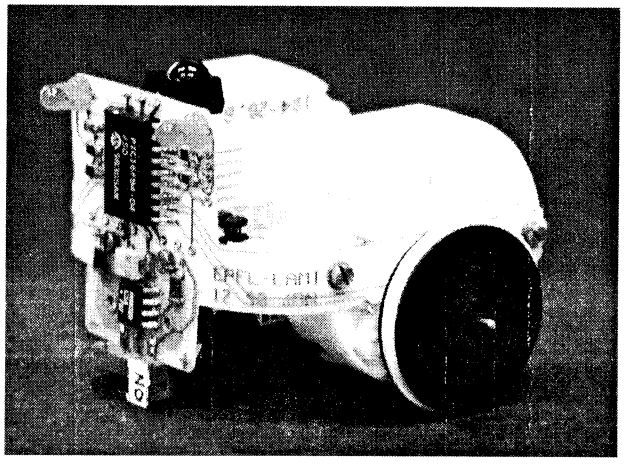


Fig 4 Swibot: a low cost 2 inch robot with space for extension modules

Sensors

A robot should recognize its environment in order to avoid obstacles and reach its goal. One widely used solution is the infrared emitter/receiver pair. Distances up to several centimeters can be measured. The maximum distance and the precision of the measure depends on the power burst on the emitting diode, which may be limited by the internal resistance of the small battery. Alice and Inchy uses side IR pairs to navigate in the maze corridor. Distance is only a few millimeters, but reliable operations are not so easy. Alice wins at Nagoya against the faster Inchy, because it was possible to adjust at the last minute the sensor sensitivity according to the reflection coefficients of the contest maze. IR pairs are more easy to implement if one just need to follow a white line on the ground.

PSD (position sensitive devices) are excellent for medium size robots (10 cm). Distances from 10 to 80 cm are measured, but the sensor may provide sometimes erratic measures out of this range. Sonars have more drawbacks.

Polarized light and vision are used by insects to navigate. Complexity is high, and miniaturized solutions, including the preprocessing will not be commercially available before several years. A small robot is the best platform to experiment and develop the technology for these new sensors, forcing the developer to take care of all the constraints of an autonomous small robot.

Ambiant light, magnetic field, odors, and sound are more or less easy to sense and amplify. A compass is convenient. An accelerometer or an inclinometer is useful only if the robot does not always stay flat on the ground.

Communication between robots and/or with a central controller can be achieved with IR or radio. Hardware may be easy to implement, but not protocols. A communication with some indication of the direction and distance to the sender is essential for insects. No good solution is available now.

Low level control

The basic behaviors given to autonomous robots is moving in a straight line, following a wall, and avoiding obstacles. If sensors are reliable, it is easy to program a simple controller for this task. For the genetic algorithms which may be used to evolve complex behaviours, it may

be required to get some measure of performance at this level, in order to calculate the fitness function.

Autonomy

The navigation task can be programmed, learned or evolved. Experiments with Khepera and other robots show that genetic evolution is a powerful mean for reaching complex behaviours. Studies in this field imply a good simulator and a reliable platform to demonstrate that the algorithms would also work with real devices. The danger with simulators is that both the world and the sensors are models. Frequent transports from the simulator to the physical platform is required to check the match between the model and the reality.

Power source

Batteries have a rather poor energy to weight ratio. It will improve over the years, but one can only hope for a factor of two. The major problem when the battery is getting small is the internal resistance. If you imagine building an autonomous 1cc robot with the 1.9 mm motor, you have to connect a small watch battery having an internal resistance of about 20 Ohm to a motor with an internal resistance of 3.5 Ohm. Transfer efficiency will only be 15%!

Supercaps (high capacity condensators) are good way to accumulate energy and restore it by burst, but again, their internal resistance decreases significantly with the size. This bad situation will make it very difficult the fly of an autonomous micro-robot, even for a few seconds.

Studying robot autonomy supposes long term experiments. Khepera can last for only 20 minutes, due to the good processor and the additional turrets. The only solution for longer operations is to use these low power watch motors. Alice and SwiBot autonomy is 8 hours. Motion speed is slow, but this leaves the robot processor time to process its sensors and think. Low power processor are available, and the choice is broadening. A well programmed PIC processors at 4MHz is amazingly powerfull. The StrongArm processor provides top performance at rather low power, with high level software support.

Automatic recharging

Even with a watch motor and a carefull processor design, batteries will have a limited life. Changing the battery and restarting the program is the usual procedure, due to the difficulty to reload the battery on the robot. Charging circuits are not so large and difficult to implement. The problem is the autonomous navigation toward the charging station, and the quality of the electrical contacts required between the power source and the robot. A fixed active mechanism that grips the robot during the charge seems to be the only reliable solution. Implementing the grabbing mechanism on the robot would be more bio-inspired, but add unuseful complexity to an already difficult to design miniature robot.

Conclusion

Much more should be said about the problems of designing miniature robots and their present and expected future solutions. Several significant technology steps are required before complex behaviour can be demonstrated, specially on the sensor side. An immediate challenge is to learn how to use low power microcontrollers for implementing behaviours in some original way. Brooks started 10 years ago to connect a bunch of HC11 to get a more complex behaviour. For the same power, is it preferable to interconnect several low frequency processors like the PICs, or one faster Scenix or StrongArm? One million transistors is possible in an 8-pin micro package. Should they be used for floating point units or for 8-bit up/down counters representing the states of a set of neurons and pointers to tables? One million neurons drives an insect. We are technologically close to that brain processing power, but there may be a long way on the algorithmic side.

References

[1] F. Mondada et al., "Mobile robot miniaturisation: A tool for investigation in control algorithms", Proceedings of the Third International Symposium on Experimental Robotics, ISER'93, Kyoto, Oct. 28-30, 1993, pp. 501-513

[2] G. Caprari et al., "The Autonomous micro robot Alice: a platform for scientific and commercial applications, Proceedings of MHS'98 conference, Nagoya, November 98, pp 231-235 and http://dmtwww.epfl.ch/~gcaprari/

[3] http://www.smoovy.com/ Fax +41 32 655-8302

[4] Control and Analysis of Low Inertia Miniature Synchronous Motors, Proceedings of MHS'98 conference, Nagoya, November 98, pp 97-103

[5] J.D. Nicoud, "Programming the Microchip-PIC microcontrolers", Internal report, LAMI-EPFL, fax +41 21 693-5263, http://diwww.epfl.ch/lami/

[6] http://www.didel.ch/ Fax +41 21 728-4483 or http://diwwww.epfl.ch/lami/mirobots/

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Theater, Movie with A-Life

-Romeo & Juliet in Hades as A-life based cinema-

Naoko Tosa
ATR Media Integration & Communication Research Laboratories
2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0288 Japan
tosa@mic.atr.co.jp

Abstract

Ever since the Limier brothers created Cinematography at the end of the 19th century, motion pictures have undergone various advances in both technology and content. Today, motion pictures, or movies, have established themselves as a composite art form in a wide domain that extends from fine arts to entertainment. A-life based Interaction technology provides movies with much greater inherent possibilities than the current forms of movies, because it allows each viewer to get involved in the movie world, metamorphose into the main character in a movie and enjoy a first-hand experience. Based on this concept, we have developed an A-life based interactive movie system. This system has two basic characteristics. The first characteristic is the use of A-life based Computer Graphics technology and the generation of three-dimensional imagery to create a autonomous actor and cyberspace in which all participants obtain a feeling of immersion. We can developed Multi-person participation, Emotions recognition, gesture recognition, the participants the feeling of actually contributing to the development of the story in the cyberspace, we use a system that shows avatars as the alter egos of the participants on a screen. Also,the system to we have produced an interactive story based on this system. We selected "Romeo and Juliet" by Shakespeare as the base story. The main plot of the story is as follows. After their tragic suicide their souls are sent to Hades, where they have no recollection of anything. Then, each of them starts on a journey to rediscover who he/she is and what relationship they shared with A-life based autonomous actors.
Keywords: film, theater, A-life,interactive movie,speech recognition, gesture recognition, emotion recogntion

1. Introduction

When humans dream during sleep, we are centrally involved in the events of our dreams. We are the lead character, and walk, talk, feel happy, feel sad, and actually even sweat. Is our consciousness active at this time? Moreover, is our deep psyche working at this time? When a novel or a film touches our emotions, our mind enters into its fantasy world. However, no matter how much we empathize with this world, we can never be more than just an outsider. I have thought what it would feel like to break down this barrier, and have started researching interactive theater using the latest technology. Interactive theater is an activity whereby one enters the film world and experience the narratives of the movie world not as an outsider but as a subject. One then tries to influence the actors and change the storyline depending on the conversations and actions that have taken place.

2. Empathy and Catharsis

In theater and film, the most difficult but valuable effect is catharsis. Defined by Aristotle, it is the soothing release of the emotions. In theater, however, not all emotions aroused by theater are happy ones. Pity, anxiety and fear are mainstays of the theatrical style. It is the release of feelings that are thought to be pleasant rather than the emotions themselves. In the specific context of dramatic activities, the arousal and release of emotions have the power to purify our soul or inner world. The Deeper empathy we feel toward the narratives of the theater and film, the higher purified state we can get. Therefore both empathy and Catharsis are essential for these media.

3. The Film Media

The events told by a film is the narratives. This tells of the relation among humans and feelings between one human being and another. These emotions are depicted by the film maker in his/her own special way. The major difference between film and theater is the camera work, the way scenes are expressed in an ordinary life way, and the fact that the time axis can be edited when working with film.Because, these factors film may be called the expression ofmemory.

4. The Theater Media

Theater styles are visualized through the patterns of emotional tension coming from the audience. In the typical form, tension increases as the play progresses and a climax in the action is reached. This is then followed by a period of calm. The climax of a play is both universal and necessary, and marks the moment when all other possible story directions are eliminated. It is for this reason that the climax is the pinnacle not only in a emotional sense, but also in an informative sense. There is a direct connection between what we find out and how we feel. The control of information establishes fate and universality, and triggers the release of tensions and emotions just at the

moment of catharsis.

5. Theater Does Not Describe

(1) Theater is performance. It is not reading; it is the expression of action. Both what we directly feel and what we understand have relevance to theater performances. Story consist of description, while theater consists of action.

(2)Theater does not express events as they occur in our dairy life. Only significant or symbolic actions/utterances are selected and presented to the audience in an exaggerated form. In other words, theater presents each event in an condensed form, thus giving strong impression to the audience. On the other hand, film expresses events as they occur in our daily life. This means that events is expressed in an diffused way in the case of film. On the other hand, film has the capability of controlling time axis more freely than theater. Film gives the audience feeling of empathy by utilizing this time control function.

(3) Theater has consistency in its action, while storytelling has a episode-like structure. For example, as each scene of film is a clip extracted from our everyday life, film is a sequence of "episode." Therefore to give audience strong impression, it is necessary for a director to utilize the contextual and temporal relationship among each scene. On the other hand, theater can show each scene as a different one from our everyday life or even as an abstract one. Therefore it is easier for a theater director to construct tight relationship among each theater scene and as the result to create the stronger effect of climax and catharsis.

6. Future Movie as a New Media

The proposed interactive theater is a medium that mixes film (prerecorded material) and live theater, and brings together the viewpoints of both those watching and those being watched (the actors). Remote performances of this type of theater across different time zones and in different languages will become possible through the use of networks.

7. Avatar

To create the impression that you are really living in the fictitious world of film, an alter ego that can reflect you as you are and can be controlled by you is necessary. The proposed avatar design concept adopts human silhouettes. However, using silhouettes alone will not produce a strong representation when shown in a three-dimensional space. Therefore, wooden marionettes on shafts have been added. The hearts of these marionettes are made of three metal cogs which only rotate when the avatars are being controlled by members of the audience. The head, body, arms and legs of each avatar are controlled by a audience member who is fitted with magnetic sensor ware. In order to make you aware of your movements, the avatar is devised to make the noise of these movement by hitting wood against metal. This is designed to make you feel like you are moving like a robot. Fig.1.



Fig.1 People play Romeo by using avatar

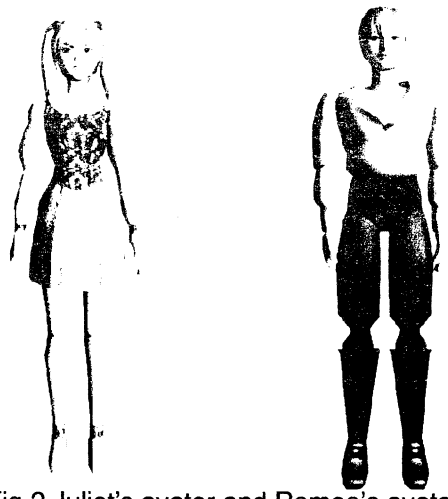


Fig.2 Juliet's avater and Romeo's avatar

8. Interactive Story

A style has been adopted that brings a story to life in a similar way to how we dream through communication between the actors and with the avatars in a film-like production. We have put together an interactive scenario that deals with the fate of Romeo and Juliet after their deaths. The theme is the sentimental relationship between people in an imaginary world.

It goes like this. Humans, like other animals, keenly pursue other lives. Despite this pursuit, however, there are many circumstances that are out of our control. Encounters and separations, whether good or bad, are strange events, and can not be accurately predicted by any person. When a computer stands between such living relationships, we must ask ourselves whether it is possible that this intervention may change these relationships.

The scenario is devised by the people playing Romeo and Juliet who select lines depending on their mood. How the drama unfolds depends on this selection process. Figures 3-5 show a number of scenes.

9. Conscious Interaction

Let's now take a look at the possibility of feeling empathy while undertaking conscious interaction. Our aim is to bring out catharsis, namely the seemingly pleasant release of emotions. However, if the techniques used for interaction do not work properly, this release will stop midway, producing unpleasant type of feelings. The in-

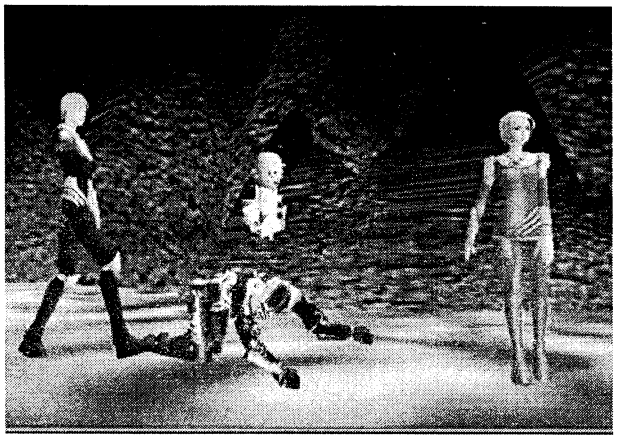


Fig.3



Fig.4

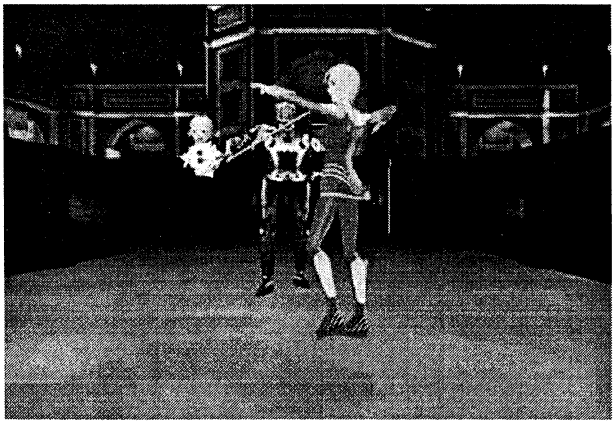


Fig.5

teraction technology used and the combinations of these patterns can greatly affect good and bad consciousness.

10. Software Realizing Conscious Interaction

(1) Anytime interaction

Anytime interaction is the design function that allows players to interact with other characters or objects in the scene at any moment during the story whether prompted by a sign, obstacle, conflict, discovery, change of heart, success or failure.

* Follow interaction: This function gives the characters in the movie system to follow or accompany the avatar of a player in an appropriate situation. For example, if the player who play the role of heroin Juliet is in a serious situation, an autonomous agent Shin, an angel, recognize her mental situation by emotion recognition, accompany, and take care of her.

* Background speech interaction: This function allows you to talk from the background to an character about your feelings and doubts concerning the performance running in the foreground.

* Touch interaction: This function is used when, for example, the lead actor Romeo becomes very angry and hits his old enemy Paris. Paris then pulls back in fear.

(2) Emotional recognition

If the people playing the roles of Romeo and Juliet speak words of anger or of happiness, depending on the context of the play, autonomous characters recognize the emotions from the tone of voice and express their emotions reactions by speech and animations.
Emotional recognition technology allows a neural network to learn emotional speech, and through this learning, the network can create personalities, such as an angry character or a cheerful character.

(3) Voice recognition is the function whereby the lines chosen by a player to go with the scene are said, and the autonomous character recognizes the meaning of these lines and reacts to these lines by their utterance and gestures.

(4) Gesture recognition is the reaction of the character to action, contact or a pose in any location. Take for example if the impassioned Romeo was to try to kill his former friend, Macutio, with a pistol. Macutio would recognize this behavior, escape and condemn Romeo.
(4) Sound output subsystem

Figure 6 shows the structure of the software used in the system.While the first system stressed story development, the system had to achieve a good balance between story development and impromptu interaction by incorporating the concept of interaction at any time. This required building a distributed control system instead of a top-down system structure.

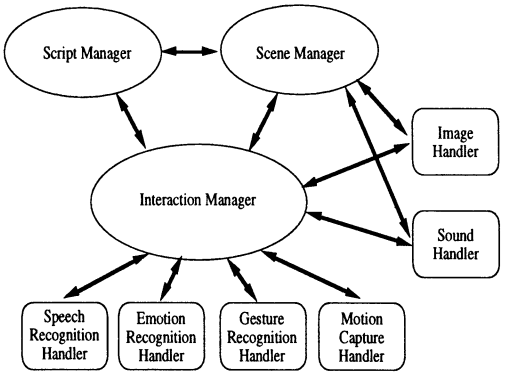


Figure 6. Software configuration of the system.

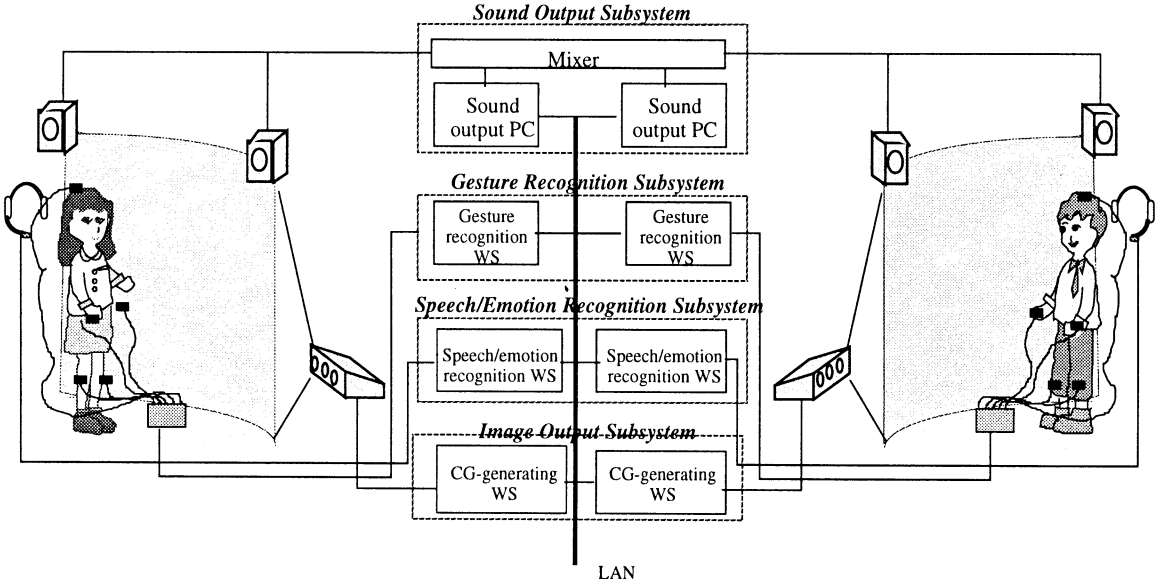


Figure 10. Hardware configuration of the Interactive cinema system

12. Hardware sytem structure

Figure 10 shows the system's hardware structure, comprised of image output, voice and emotion recognition, gesture recognition and sound output subsystems.

(1) Image output subsystem
Two workstations (Onyx Infinite Reality and Indigo 2 Impact) capable of generating computer graphics at high speed are used to output images. The Onyx workstation is used to run the script manager, scene manager, interaction manager and all image output software. Character images are stored on the workstations ahead of time in the form of computer graphic animation data in order to generate computer graphics in real time. Background computer graphic images are also stored as digital data so background images can be generated in real time. Some background images are real photographic images stored on an external laser disc. The multiple character computer graphics, background computer graphics and background photographic images are processed simultaneously through video boards on both the Onyx and Indigo 2 workstations. Computer graphics are displayed in 3-D for more realistic images, and a curved screen is used to envelop the player with images and immerse the player in the interactive movie world. Image data for the left and right eye, created on the workstations ahead of time, are integrated by stereoscopic vision control and projected on a curved screen with two projectors (Fig. 11.).

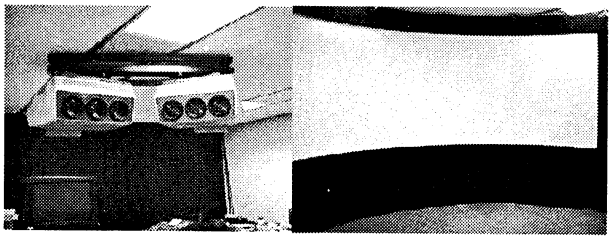


Figure 11. Projectors and arch screen

13. Future Work

Future topics include describing interactive situations to improve the real-to-life feel of each scene. Here I do not mean a realism that blurs the line between what is real and what is not. What I envisage is the generation of unknown pleasant imagination that lays between people. This experience does not necessarily have a happy ending. The independent actors may betray the humans, while the humans may grumble about their problems, be afraid or feel pity. After this experience, however, we will have a digital catharsis in which a sense of fulfillment can be felt. To raise improve this cathartic feeling, we will need to use the most advanced technology to express the context of each scene. Ultimately this is linked to research on the unknown elements of humans such as the boundary between consciousness and unconsciousness.

14. Conclusion

Machines become most beautiful when they most resemble living forms. Even computer graphic actors are touched when something human-like is felt. In our age of machines, visual experiences using the most advanced technology are probably the closest to this feeling in chines. We have already come across unimaginable new man-made beauty in films using computer graphics. If we move from the era of machines to the era of images, it can be said that we will bring forth a new world of consciousness deeply related to the imagination. This will generate a new consciousness for humans that communicate with machines. This may even be experienced through the up and coming Future Movie .

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Treatment of Nonverbal Behaviors for Alife-based Computer Actors

Ryohei Nakatsu, Alexandre Solomides, Naoko Tosa
ATR Media Integration & Communications Research Laboratories
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, Japan

Abstract

Nonverbal behaviors such as recognition/generation of emotion plays an essential role in human communication. In the future, it is expected that computer characters will have the capability of treating nonverbal as well as verbal communication behaviors. In this paper, we first study the recognition of emotions involved in human speech. We propose an emotion recognition algorithm based on a neural network and also propose a method to collect a large speech database that contains emotions. We carried out emotion recognition experiments based on the neural network trained using this database. An emotion recognition rate of approximately 50% was obtained in a speaker-independent mode for eight emotion states.

We then study the nonverbal behavior generation mechanism of computer character in the interactive movie system we are developing. We learned from our first prototype system that interactive movie characters should achieve nonverbal behaviors such as spontaneous reactions. We propose to use emotion recognition as key technology for an architecture of the computer characters with spontaneous behaviors.

1. Introduction

Nonverbal behavior plays a very important role in human communication. Telephones have been mainly used to communicate in business, but recently, telephones are used more and more for everyday communication among family members and friends. The spread of cellular phones, especially among the young generation, has accelerated this trend. It is clear that the nonverbal behavior such as the exchange of emotions is important in these forms of communication and is sometimes more important than verbal behavior. This means that nonverbal behavior is the basis of human communication.

In addition to human-to-human communication, communication between human and computer agents has becomes more and more common. Computer agents that act as electronic secretaries or communication mediators will become common entities in our society. As such, the capability of communicating with humans based on both verbal and nonverbal behaviors will be essential. This will surely make interactions between computers and humans more intimate and human-like.

Although the importance of nonverbal behavior in communication has been recognized, until now most research has involved nonverbal behavior based on images. Facial expression recognition and gesture recognition are good examples. On the other hand, the recognition of emotions involved in human speech has been rarely treated. For the reasons indicated above, we have studied the recognition of emotions involved in speech and believe such recognition is an essential research area.

The main reason why there has been little research on human emotion recognition is because it is difficult to collect a large amount of utterances that contain emotions. The strategy we have adopted here is to ask a radio actor to utter a number of words with various kinds of emotional expressions. Then, we asked many speakers to utter utterances with emotion by listening to the utterances uttered by the actor. We adopted eight emotions including a normal state. We succeeded in collecting a large speech database uttered by fifty males with this method . By using a part of this database as a training data for neural network, we obtained a neural network which can recognize emotional utterances. Then, we carried out the recognition experiment to evaluate the performance of the neural network. We compared two kinds of recognition experiments, open test and closed test, and concluded that a recognition rate of approximately 50% is obtainable for speaker-independent emotion recognition.

For the next step, we applied this emotion recognition technology to computer characters that can communicate with humans based on verbal and nonverbal behaviors. We have conducted research on interactive movie production by applying interaction technologies to conventional movie making techniques. The integration of interaction and narratives is expected to produce a new type of experience, which we call, "Interactive Movies." We can interact in as well as watch the story in interactive movies. This gives us a great opportunity to learn various kinds of skills and lessons through dramatic experiences. We have already produced a prototype system [1]. Unfortunately, in this system computer characters lack the capability of nonverbal behavior such as spontaneous behaviors. In evaluations, we have learned that spontaneous interaction is the key element for subject participation in narratives. We developed a second prototype system based on this evaluation where emotion recognition works as a key function for realizing computer characters with spontaneous behavior capabilities.

This paper first describes the emotion recognition algorithm and the emotion recognition experiment we have carried out. Then, the paper introduces the configuration of the interactive movie system and the structure of the computer characters in the system where emotion recognition plays a key role for the introduction of spontaneous behavior capabilities.

2. Recognition of Emotions Involved in Speech

2.1 Basic principle

We have considered and emphasized the following issues to recognize emotions.

(1) Treatment of various emotional expressions

How many and what kinds of emotional expressions are to be treated are interesting yet difficult issues. The following are some examples of emotional expressions treated in several papers:

a. neutrality, joy, boredom, sadness, anger, fear, and indignation [2]

b. anger, fear, sadness, joy, and disgust [3]

c. neutrality, happiness, sadness, anger, fear, boredom, and disgust [4]

d. fear, anger, sadness, and happiness [5]

Considering these examples, we have selected eight emotional states in this study:

anger, sadness, happiness, fear, surprise, disgust, playfulness, and neutrality

(2) Speech features

There are two kinds of speech features: phonetic features and prosodic features. In emotion recognition, prosodic features play an important role. At the same time, phonetic features are as important as prosodic features, because prosodic features and phonetic features are tightly combined when uttering speech. Furthermore, it is impossible to express emotions by only controlling prosodic features. Therefore, a combination of two kinds of features is considered in this study: one is the feature expressing phonetic characteristics of speech, and the other is that expressing prosodic characteristics.

(3) Speaker-independent and content-independent emotion recognition

Speaker independence is an important aspect of speech/emotion recognition. From a pragmatic standpoint, a speaker-dependent emotion recognition system requires a tiresome learning stage each time a new speaker wants to use the system, so it is not easy to use. Another point is that humans can understand the emotions included in speech as well as the conveyed meaning by speech even for arbitrary speakers. Moreover, content independence is indispensable for emotion recognition. Various kinds of emotions are conveyed for the same words or sentences in daily communication; this is the key to rich and sensitive interpersonal communications. Thus, we adopt a neural network architecture and introduce a training stage that uses a large number of training utterances for a speaker-independent and content-independent emotion recognition system.

Figure 1 illustrates a block diagram of the processing flow. The process mainly consists of two parts: speech processing and emotion recognition. The details of each process and the system configuration for carrying out the emotion recognition process are described in the following sections.

2.2 Feature extraction

(1) Speech feature calculation

Two kinds of features are used in emotion recognition. One is a phonetic feature and the other is a prosodic feature. LPC (linear predictive coding) parameters [6], which are

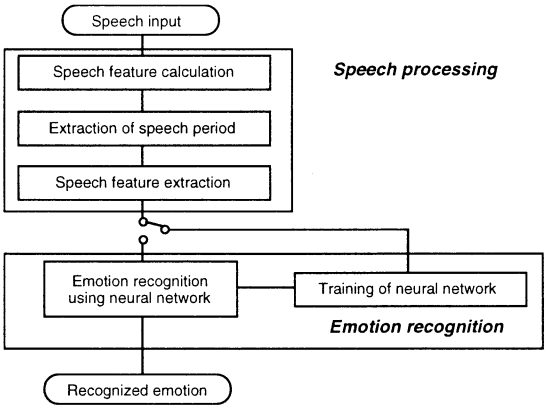


Fig. 1 Processing flow diagram.

typical speech feature parameters often used for speech recognition, are adopted for the phonetic feature. The prosodic feature, on the other hand, consists of three factors: amplitude structure, temporal structure and pitch structure. Speech power and pitch parameters are used for the feature expressing amplitude structure and pitch structure and each can be obtained in the LPC analysis. In addition, a delta LPC parameter is adopted, which is calculated from LPC parameters and expresses a time variable feature of the speech spectrum, since this parameter corresponds to a temporal structure.

The speech feature calculation is carried out in the following way. Analog speech is first transformed into digital speech by passing it through a 6 kHz low-pass filter that is then fed into an A/D converter with an 11 KHz sampling rate and a 16 bits accuracy. The digitized speech is then arranged into a series of frames, where each is a set of 256 consecutive sampled data points. LPC analysis is carried out in real time and the following feature parameters are obtained for each of these frames.

Speech power: P
Pitch: p
LPC parameters: c1, c2, ..., c12
Delta LPC parameter: d

Thus for the t-th frame, the obtained feature parameters can be expressed by

$$Ft = (Pt, pt, dt, c1t, c2t, ..., c12t).$$

The sequence of this feature vector is fed into the speech period extraction stage.

(2) Speech period extraction and speech feature extraction

First, the period where speech exists is extracted based on the information of speech power. Speech power is compared with a predetermined threshold value PTH; if the input speech power exceeds this threshold value for a few consecutive frames, the speech is decided to be uttered. After the beginning of the speech period, the input speech power is also compared with the PTH value; if the speech power is continuously below PTH for another few consecutive frames, the speech is decided to be no longer exist. The speech period is extracted from the whole data input through this process.

Twenty frames are extracted for the extracted speech period where each is situated periodically in the whole speech period and kept the same distance from adjacent frames. Let these twenty frames be expressed as f1, f2, ..., f20. The feature parameters of these twenty frames are collected and the output speech features are determined as a 300 (15x20) dimensional feature vector. This feature vector is expressed as

$$FV = (F1, F2, ..., F20),$$

where Fi is a vector of the fifteen feature parameters corresponding to frame fi.

This feature vector (FV) is then used as input for the emotion recognition stage.

2.3 Emotion recognition

Recognizing emotions is a difficult task. The main reason is that people mainly rely on meaning recognition in daily communication, especially in business communication. This is why speech recognition research has long treated emotions contained in speech as simply fluctuations or noise. What makes the situation more complicated is that emotional expressions are consciously or unconsciously intertwined with the meaning of speech. In the unconscious state, context rather than emotional feature plays a more important role. As a result, the intensity of emotional expression varies dramatically depending on the situation. Of course, our final target is to recognize emotions in speech even if emotional expression is unconsciously mixed with the meaning of speech. However, for the time being, this is not our research target for the above reasons. Instead, the strategy adopted here is to treat speech intentionally uttered with specific emotional expressions, rather than speech with unconscious emotion expressions.

There are several algorithms such as neural network or HMMs[7] in recognition algorithms. HMMs are suitable where the structure of the recognition object is clear to some extent. As the structure of an emotional feature is not clear, a neural network approach seems more suitable, so we have adopted the neural network approach here.

(1) Configuration of the neural network

The configuration of the neural network for emotion recognition is shown in Fig. 2. This network is a combination of eight sub-networks. Each of these eight sub-networks is tuned to recognize one of eight emotions (anger, sadness, happiness, fear, surprise, disgust, playfulness, or neutrality). The construction of each sub-network is shown in (Fig. 3) and basically has the same network architecture. It is a four layered neural network with one 300 input nodes corresponding to the dimension of speech features and 1 output node. The number of intermediate nodes varies depending on the specific emotion. The reason we have adopted this architecture is based on a consideration of the difficulties in recognizing specific emotions. Thus, it is easier to prepare a specific neural network for each emotion and tune each network depending on the characteristics of each emotion to be recognized. This basic consideration was confirmed by carrying out preliminary recognition experiments. Although negative emotions such as anger or sadness are rather easy to recognize, positive emotions such as happiness can be difficult to recognize. Thus, a detailed architecture of the networks, such as the number of intermediate nodes, differs depending on the specific emotion.

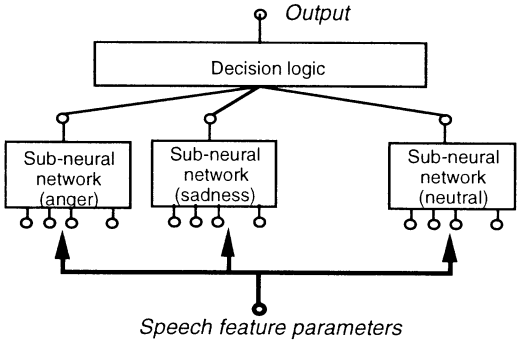


Fig. 2 Emotion recognition part configuration.

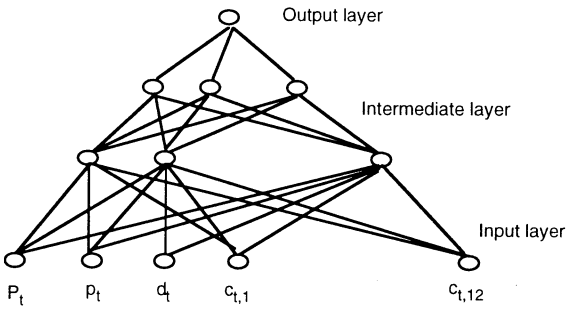


Fig. 3 Sub-network configuration.

(2) Emotion recognition by a neural network

In the emotion recognition phase, speech feature parameters extracted in the speech processing part are simultaneously fed into the eight sub-networks and trained as described in the above process. Eight values, V=(v1, v2, ..., v8), are obtained as the result of the emotion recognition.

2.4 Emotion recognition experiment

(1) Speech database collection

It is necessary to train each of the sub-networks for the recognition of emotions. The most important and most difficult issue for neural network training is how to collect a large amount of speech data containing emotions. As our target is content-independent emotion recognition, we adopted one hundred phoneme balanced words for a training word set. Some examples are: school, hospital, standard, and so on. Since we utter most of these words without any special emotion in our daily life, it is difficult for ordinary people to intentionally utter them with emotions. Therefore, we have adopted the following strategy.

(a) First we ask a radio actor to utter one hundred words with each of the eight emotions. As a professional, he is used to speaking various kinds of words, phrases, and sentences with intentional emotions.

(b) Then, we ask speakers to listen to each of these utterances and mimic the tones of each utterance. We record the utterances spoken by ordinary people.

The problem with this strategy is that the spoken emo-

tions here are not natural but "forced emotions." However, we study the forced emotions or intentional emotions in our research based upon the consideration described in Section 2.3.

Since our target is speaker-independent and content-independent emotion recognition, the following utterances were prepared for the training process

Words: 100 phoneme-balanced words
Speakers: fifty male speakers
Emotions: neutrality, anger, sadness, happiness, fear, surprise, disgust, and playfulness
Utterances: Each speaker uttered 100 words eight times. In each of the 8 trials, he/she uttered words using different emotional expressions with a total of 800 utterances for each speaker obtained as training data.

(2) Training and recognition experiment

We used thirty speakers for training out of the fifty speakers used for data collection. To learn the effect of the number of speakers used for the training, we carried out five types of neural net trainings and obtained the following neural networks.

(neural network 1) ten types of neural networks each trained by a single speaker (#1, #2, .., #10).

(neural network 2) five types of neural networks each trained using two speakers (#1and #2, #3and#4,..., #9and#10)

(neural network 3) two types of neural networks each trained using five speakers (#1and#2..and#5, #6and#7...and#10)

(neural network 4) a neural network trained using ten speakers (#1and #2and #3... and #10)

(neural network 5) a neural network trained using thirty speakers (#1and #2and #3 ... and #30)

In addition, we carried out two types of recognition experiment to evaluate the performance of the obtained neural networks.

(Open recognition experiment)

In this case, utterances spoken by the speakers not included in the training sets are used for the recognition experiment. Twenty speakers (#31-#50) were used for the recognition experiment.

(Closed recognition experiment)

In this case, utterances spoken by the speakers included in the training sets are used for the recognition experiment.

The obtained recognition results for both closed recognition and open recognition are shown in Fig. 4. These results show the following facts.

(a) For closed recognition experiments, the recognition rate decreases as the number of training speakers increases. The recognition rate approaches 50-55%.

(b) For open recognition experiments, the recognition rate increases as the number of training speakers increases. The recognition rate approaches 50-55%.

These two trends indicate that if we have enough speakers for training we can obtain an emotion recognition rate of 50-55% for the speaker-independent mode. Furthermore, even when the number of the speaker is thirty, we have an approximately 50% emotion recognition rate that is satisfactory compared with the expected recognition performance using an adequate number of training speakers.

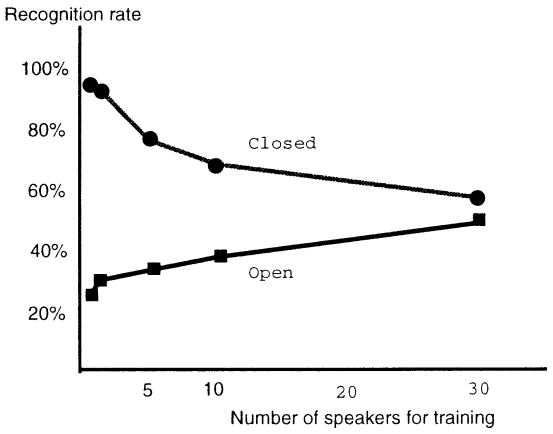


Fig. 4 Emotion recognition results.

We have concluded through these recognition experiments that we have the emotion recognition capability whereby computer characters can communicate with people based on nonverbal behaviors.

3. Creation of Computer Characters with Nonverbal Behavior Capabilities

3.1 Overview

As one of the applications of emotion recognition technology, we have tried to apply this technology to the computer character in an interactive movie system we are studying. The main reason why we study interactive movies is as follows.

Ever since the Lumiere brothers created cinematography at the end of the 19th century, movies have undergone various advances in technology and content. Today, movies have established themselves as a composite art form covering a wide range from fine arts to mass entertainment. Perhaps, movies provide us with the feeling as though we are experiencing various kinds of dramatic events and happenings in movie narratives. However, these experiences are not active and are illusions. As a result, what we can experience, feel, and learn is limited.

The integration of interaction and narratives is expected to produce a new type of experience. We call this, "Interactive Movies" , in which we not only can watch the story but also can interact in the story itself. This provides us with a totally new type of experience. In other words, we can experience dramatic events or narratives that cannot be encountered in our daily lives as a subject of the event, instead of through the perspective of a third person. This gives us a great opportunity to learn various kinds of skills and lessons through dramatic experiences.

One of the key factors of an interactive movie system is the computer characters that interact with the participants who play the main characters. In the first system we have developed, the behaviors of the characters are controlled based on the narratives [1]. In other words they achieve only verbal behaviors. We have learned in evaluating the

first system that the spontaneous behavior of the characters are as important as the narrative-based behaviors. Therefore, we have tried to integrate the spontaneous behavior capabilities into the movie characters by using emotion recognition.

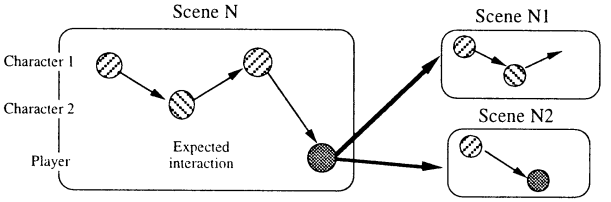
3.2 Spontaneous Behaviors

In evaluating the first prototype system, we recognized that there are generally two types of behaviors for the computer characters: narrative-based and spontaneous behaviors. We also recognized that in the first system only the narrative-based behavior capabilities were realized.

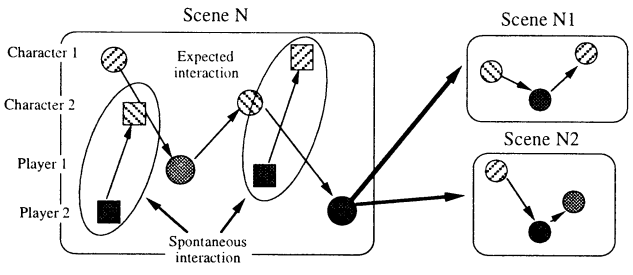
Basically, spontaneous interactions occur between the participants and characters and basically do not affect story development. On the other hand, there are times when interactions do affect story development. Such interactions occur at branch points in the story, and they tend to determine the future development of the story. The key point is how to handle these two different types of interactions.

Figure 5 illustrates how interaction proceeds for both the first system and the second system. In the first system, the order of all behaviors of a participant and behaviors of characters are predetermined. Thus, the system control mechanism is rather simple.

The difference between the first system and the second system, as illustrated in Fig. 5, is that in addition to the predetermined sequence of interactions between the participants and the characters, unpredictable interactions, in other words, spontaneous interactions, occur. Therefore, the system is required to distinguish the predicted and unpredicted inputs and handle the behaviors of the characters according to these two kinds of inputs. Furthermore, some fluctuations are added to the response of each character to add naturalness to the reactions of the characters. These requirements have led to a distributed control structure instead of a top-down structure. Figure 6 illustrates the software used in the second system. The interaction manager and the scene manager in Fig. 6 are the key components controlling the behavior of the computer characters.



(a) The first prototype system



(b) The second prototype system

Fig. 5 Comparison of interactions between first system and second system.

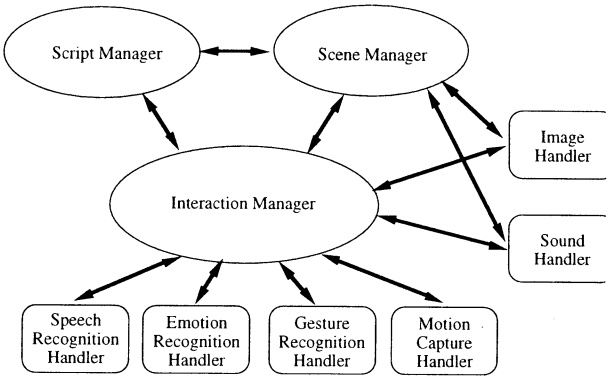


Fig. 6 Software configuration of second system.

3.3 Interaction Manager

The interaction manager is the most critical component for achieving the spontaneous behavior of the characters. The interaction manager has functions for distinguishing predicted and spontaneous inputs from the participant and generating character reactions for spontaneous input. The details of these functions are described below.

(1) Distinguishing predicted and spontaneous inputs

It is difficult for the system to distinguish whether the speech of a participant is a predicted input or spontaneous input when only using speech recognition technologies. To solve this problem, we have adopted a method in which both speech recognition and emotion recognition work simultaneously. When a reasonable speech recognition result is obtained, the input is judged as a predicted one and the recognition result is utilized for generating the predetermined behaviors of the characters. When the speech recognition function fails to output a reasonable recognition result, it is judged as a spontaneous one, and the emotion recognition result is then utilized to generate the spontaneous reactions of the characters. For gesture recognition, it is rather easy to distinguish between two kinds of gesture inputs depending on the scene because the usage of predetermined gestures is restricted.

(2) Generating character reactions for spontaneous inputs

The basis of spontaneous interaction is a structure that allots each character an emotional state, and the input from the participant and his/her interaction with the characters determine the emotional state (as well as the response to that emotional state) for each character. Some leeway is given to how a response is expressed depending on a character's personality and circumstances. To achieve this, we chose to use the concept of the "action selection network" [8], which sends and receives activation levels among multiple nodes. The interaction manager works with spontaneous inputs as outlined below.

The state and intensity of a player's (i = 1, 2...) emotion at time T is defined as

Ep (i, T), sp(i, T) where sp(i, T) = 0 or 1

(0 indicates no input and 1 indicates an input).

Similarly, the state and intensity of a character's (i = 1, 2...) emotion at time T is defined as

Eo (i, T), so(i, T).

For the sake of simplicity, the emotional state of a character is determined by the player's emotional state, when the player's interaction results from emotion recognition:

$\{Ep(i, T)\} \rightarrow \{Eo(j, T + 1)\}$.

Activation levels are sent to each character when emotion recognition results are input as

$sp(i, T) \rightarrow sp(i, j, T)$,

where sp(i, j, T) is the activation level sent to character j when the emotion of player i is recognized. The activation level for character j is the total of all activation levels received by the character:

$so(j, T + 1) = \sum sp(i, j, T)$.

A character that exceeds the activation threshold performs action Ao(i, T) based on an emotional state. More specifically, this action involves a character's movement and speech as a reaction to the emotional state of the player. At the same time, activation levels so(i, j, T) are sent to other characters:

if $so(i, T) \rightarrow THi$

then $Eo(i, T) \rightarrow Ao(i, T)$, $Eo(i, T) \rightarrow so(i, j, T)$

$so(j, T + 1) = \sum so(i, j, T)$.

This mechanism creates interaction between characters and enables more diverse interaction than simple interaction involving a one-to-one correspondence between emotion recognition results and character reactions.

(3) Other issues

(a) Time control

A difficult issue in handling spontaneous interaction is that once we permit it, controlling the time schedule for the sequence of predetermined interactions becomes difficult because the scenario as a whole is controlled by the scene data handled by the scene manager. As one solution to this problem, we introduced the concept of the "relative time counter". Here, the timer stops counting while characters are showing reactions corresponding to spontaneous input. This means that as long as the participants continue to enjoy the spontaneous interaction, the story stops proceeding. This mechanism allows the system to go to any point between a fully spontaneous interaction system and a fully narratively controlled system.

(b) Reaction collision

There are times when a spontaneous input comes in while a character is reacting to an expected player input. As a result, a collision occurs between the expected reaction and the reaction to the spontaneous input.

We introduce two modes for this situation: real-time reaction and delayed reaction. In real-time reaction, the character puts the reaction on hold and instead shows the spontaneous reaction. In delayed reaction, the character continues its behavior and after finishing it starts its reaction for the spontaneous input.

3.4 Scene Manager

The scene manager controls the behavior of the computer characters as well as the generation of scenes. To control the ongoing progress of each scene, we define the scene data, which controls all of the events for each scene. A brief description of the scene data construction is as follows.

The scene data consists of the following kinds of commands and parameters.

***MACRO COMMAND

COMMAND

Parameter

Macro commands control the description of a scene. The macro commands we have prepared are the following.

***SCENE: command to define the scene number

***TIMER: command to define the maximum duration time for each scene

***CAMERA: command to define the camera position for each scene

Commands define and describe the details of events in each scene. The main commands for the present system are the following.

BGIMG: command to define the background image

SOUND: command to define the background music and sound effects

OBJECT: command to define the CG corresponding to various kinds of objects and characters

MOTION: command to define the motion capture mechanism

INTERACTION_S: command to define the interaction based on speech recognition

INTERACTION_G: command to define the interaction based on gesture recognition

INTERACTION_E: command to define the interaction based on emotion recognition

Parameters define various kinds of conditions for each command. An example of a scene data description for a spontaneous interaction is as follows

INTERACTION_E

Character_N	Start_time	End_time
Participant_M1	Wait	File_CG(M1) File_speech(M1)
Participant_M2	Immed.	File_CG(M2) File_Speech(M2)

This means that during the time between the Start_time and the End_time, the character indicated by N can accept spontaneous inputs from participants M1 and M2. When reactions are activated based on the mechanism mentioned in (3), the reaction to participant M1 is expressed by File_CG(M1) for animation and File_speech(M1) for speech. In addition, the type of reaction is defined by Wait or Immed. for delayed and immediate reactions described in (3), respectively.

Figure 7 illustrates how the interaction manager and scene manager work, and as a result, how the interaction between the characters and participants proceeds.

3.5 Examples of Interactions

(1) An interactive story

We have produced an interactive story based on Shakespeare's "Romeo and Juliet". We chose it because it is a very well known story and people have a strong desire to act out the role of hero or heroine. We expect people to easily get involved in and experience the story.

The main plot of the story is as follows. After their tragic suicide the lovers' souls are sent to Hades, where they find they have totally lost their memory. The two start a journey to rediscover who they are and what their relationship

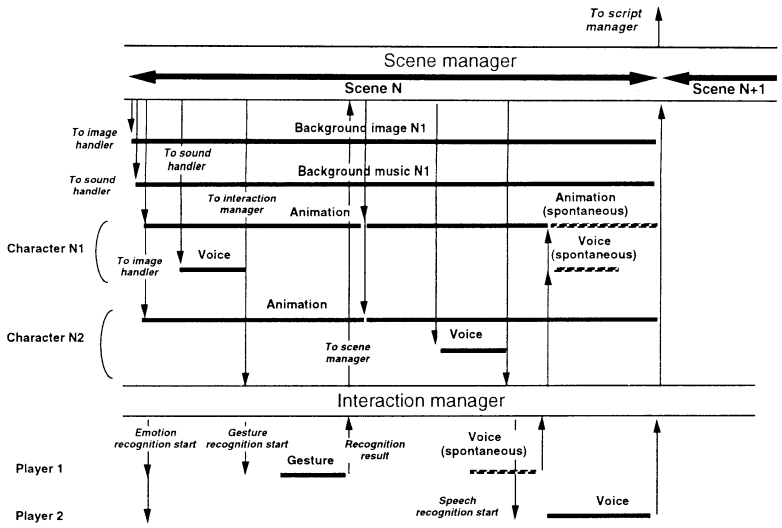
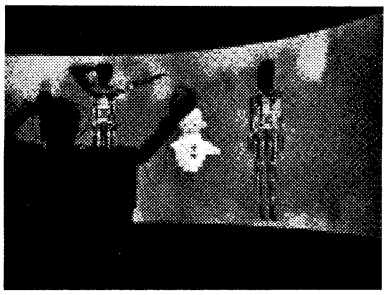


Fig. 7 Time sequence of scene/interaction manager processing (second prototype system)



(a) "Romeo" controls his avatar.



(b) 'Romeo" tries to touch object in a gift shop.

Fig. 8 Examples of interaction between participant and system.

was. They gradually find themselves again through various kinds of experiences and with the help and guidance of characters in Hades and finally go back to the real world.

(2) Interaction

There are two participants, one plays the role of Romeo and the other Juliet. The participants stand in front of the screen wearing specially designed clothes with attached magnetic sensors and microphones. Their avatars are on the screen and move according to their actions. Basically, the system controls the progress of the story with character animations and character dialogues. The story moves on depending on the voice and gestures of the participants and as described before, interaction is possible at any time. When the participants utter spontaneous phrases or sentences, the characters react according to the emotion recognition results. Consequently, this system can go anywhere between story-dominant operation and impromptu interaction-dominant operation depending on the frequency of the interaction. Figure 8 illustrates typical interactions between the participants and the system.

4. Conclusions

In this paper, we first proposed an algorithm for recognizing emotions contained in human speech. We adopted a neural network for recognition algorithm. In order to realize speaker-independent and content-independent emotions recognition, we selected a word set that consists of one hundred phoneme-balanced words. In addition, we collected utterances of these words spoken by fifty male speakers. We used this database for the training and recognition samples and obtained about 50% recognition rate in the recognition experiment.

We then applied this emotion recognition algorithm to the computer characters in our interactive movie system. In the evaluation of the first prototype system, we learned that the ability of reacting to spontaneous input by users is the key factor for realizing realistic computer characters. By utilizing emotion recognition as the key function for spontaneous reactions, we succeeded in realizing computer characters with the spontaneous behaviors.

Further research will be necessary to evaluate the emotion recognition algorithm by collecting a speech database uttered by females and carrying out recognition experiments. It also is necessary to evaluate the approach based on the integration of narrative-based and spontaneous interaction by letting people experience the interactive movie story and become main actors or actresses.

References

[1] R. Nakatsu, N. Tosa and T. Ochi, "Construction of Interactive Movie System for Multi-Person Participation," Proceedings of the International Conference on Multimedia Computing and Systems '98, pp. 228-232 (1998).

[2] S. Mozziconacci, "Pitch Variations and Emotions in Speech," ICPhS 95 Vol. 1, p. 178 (1995).

[3] K. R. Scherer, "How Emotion is Expressed in Speech and Singing," ICPhS 95, Vol. 3, p. 90 (1995).

[4] G. Klasmeyer and W. F. Sendlmeier, "Objective Voice Parameters to Characterize the Emotional Content in Speech," ICPhS 95, Vol. 1, p. 182 (1995)

[5] S. McGilloway, R. Cowie, and E. D. Cowie, "Prosodic Signs of Emotion in Speech: Preliminary Results from a New Technique for Automatic Statistical Analysis," ICPhS, Vol. 1, p. 250 (1995).

[6] J. D. Markel and A. H. Gray, "Linear Prediction of Speech," Springer-Verlag (1976).

[7] T. Shimizu et al., "Spontaneous Dialogue Speech Recognition Using Cross-Word Context Constrained Word Graph," Proceedings of ICASSP'96, Vol. 1, pp. 145-148 (1996).

[8] Pattie Maes, "How to Do the Right Thing," Connection Science, Vol. 1, No. 3, pp. 291-323 (1989).

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Technoetic Theatre: performance and enactment in the dramaturgy of artificial life.

Roy Ascott
University of Plymouth and
University of Wales College, Newport
caiia-star.newport.plymouth.ac.uk
Mailing address: 64 Upper Cheltenham Place Bristol BS6 5HR England

Abstract

This paper examines the implications of Alife for the generation of a new kind of theatre. Traditionally, Western Theatre (which in this context includes both filmed and live production) has always communicated within a fixed framework to a more or less passive audience. In this form of theatre, the transmission of meaning is controlled centrally by a director, the supreme authority, for whom the actor is the instrument of representation. Closure and resolution mark this dramatic form in which audience interaction is seldom invoked, and rarely utilised to affect the narrative's structured catharsis or resolution. Audience participation is usually illusory.

In addressing the theoretical aspects of Alife and theatre, the paper emphasizes the importance of "enactment" over "performance" within the public dimension of a work. Instead of interpretation and representation, technoetic theatre will be concerned with ideas of emergence and construction in the generation of meaning and in the reframing of consciousness.

Introduction

For the purposes of this paper, the term "theatre" is intended to refer to both cinematic and live dramatic production, while recognising the important difference between the two existing forms. In fact cinema is even less "interactive" than the stage, since live actors often claim some degree of interactivity with, or at least responsiveness towards, their live audience. In the traditional movie this is in no way possible. However, relative to the issues of interactivity and emergence that this paper discusses, the differences between these two traditional dramatic genres are less important than their commonalties. Thus this paper will employ "theatre" as a generic term.

This paper examines the theoretical implications of A life for the generation of a new kind of theatre, which we shall call "technoetic theatre", as it arises from the confluence of researches into artificial life, interactive drama and the field of consciousness. Western theatre has characteristically relied on a linear structure that seeks to elicit emotions in, and to convey meaning to, an audience whose role is essentially receptive, set within a framework which is both explicit and constrained. The relationship between the viewer and the viewed is fixed. The construction of meaning and emotion is controlled centrally by the director for whom the actor is the instrument of a determinative representation. The actor's performance is intended interpret a role within a communicative continuum, which is essentially one way: feedback from the audience serving only in the reinforcement of this objective. Closure and resolution mark the dramatic form.

In live theatre, there may be exceptionally some interaction between the audience and the dramatic construction but rarely in a way that influences its intended catharsis or the narrative trajectory. The dramatic ensemble is an object to be viewed at a distance within a strict economy of space and time. Theatre is thus rarely open-ended or uncertain; the interchangeability of roles between the viewer and the viewed is restricted. Audience participation is generally illusory, often no more than a dramatic conceit intended to flatter the viewer with an apparent transfer of power which is in fact no more than its reinforcement in the hands of the director. This form of theatre is powerful in conveying meaning and in reinforcing values and attitudes.

This paper argues that Alife, by contrast, suggest the possibility of an entirely different kind of theatre whose attributes and objectives would radically change our view of its meaning and

function. While the principles of emergent form and total interactivity will be discussed, the paper will focus on the necessity of recognising the importance of "enactment" over "performance" in this context. Instead of the classical focus on interpretation and representation, this form of theatre will be concerned with ideas of emergence and construction in the generation of meaning and in the reframing of consciousness.

Current research

This paper is propositional and speculative, a form necessitated by the present artistic and technological relationships between the movie, theatre and Alife, which can only properly be described as nascent when compared to the legitimate goal and ambitions of current practitioners. However, this is not to say that there is no significant research being pursued in this conjunction, or that the magnitude of these issues is underestimated in the commercial and cultural fields they will eventually affect. The contrary is the case. For example, in Japan at ATR significant attention is paid to this artistic (and ultimately commercial) emerging field, as evidenced in its support of the research of such artists as Naoko Tosa and Christa Sommerer and Laurent Mignonneau.

While Tosa's "interactive movies" explicitly deal with dramaturgical processes taking place in the interspace between Alife action and realtime human intervention, Sommerer and Mignonneau are concerned with Alife as an extension of fine art practice, interactive virtual sculpture or scenarios of autoperformance art. It can be argued however that their work too has affinities with cinema While Naoko Tosa's work extends and liberates the frame of the narrative movie, Sommerer and Mignonneau can be seen as revising the precepts of structuralist film, placing the viewer at the centre of the structural process. If we were to conduct a survey of this emergent field we would find a huge variety of artistic and commercial trajectories and practices. At the same time there would be very little literature to serve as a theoretical underpinning.

Theory

Hitherto, little attention has been paid to the theoretical implications of the convergence of interactive drama and Alife, largely because the field is at a very early stage of development. However the very absence of such theory invites speculation, and it has been the anticipatory, if not visionary, speculation by the Alife community from its inception that has fired the interest of artists and writers.

What does it mean to say that the convergence of Alife and Theatre would radically change our view of the meaning and function of dramaturgy? Firstly, it must be said that the theory of Alife far outstrips its present realisation. Whether based in a silicon substrate or within the substance of a wet, or at least 'moist' biology, Alife is at best rudimentary set beside its aim to produce life-as-it-might-be. Nevertheless, the aspirations of Alife and the theoretical speculation it has engendered have radically transformed the way we view "theatre" and the evolution of movies, games and other dramatic structures. This consequence however may be of less importance than its potential to radically change our view of ourselves. The theatre has always had the dual role of both reflecting ourselves as we see our selves and eliciting in us new forms of behaviour, and extending the range of our emotions.

Such also will be the case with technoetic theatre. However, it will reflect a wholly new human scenario, a completely different view not only of the human individual but also of life itself. The effect of Alife theory on thinking about theatre can be to set up a route, indirectly but significantly, to the realisation of "life-as-it-could-be". In this respect technoetic theatre should properly be seen as post-biological, and an important part of the larger cultural discourse surrounding technology, artifice and nature.

It is the emergent properties of Alife, the capacity for autoconstruction, and eventually perhaps autopoeisis, that gives it special significance in the new prospectus for interactive drama. The technoetic theatre will define a form of theatre intrinsically concerned with ideas of emergence and construction, particularly in the generation of meaning and form. In this way it can be said that Alife interactive drama can play a decisive role in the reframing of consciousness. The shaping of consciousness has long been the purpose of traditional theatre: the wish to tell a story and thereby affect morally, emotionally or politically, the mind of the audience. The Alife ingredients of technoetic theatre however will mean that while the context is provided ahead of time, the content is wholly and decisively created by the audiences

interaction with the artificially living elements of an emergent scenario.

However, while the principles of emergent form and total interactivity are of central relevance to these issues, and the current fascination with intelligent agents and avatars should not be underestimated, this paper will focus on the audience, or rather its necessary absence, and will argue the importance of "enactment" over "performance" in this context. Because interpretation and representation have always been the principal vehicles of drama, the idea of performance characterising the actor's role has rarely been challenged. But Alife can have nothing to do with interpretation or representation, unless it is to be relegated to the production of mere effects.

Enactment

It can be argued that the roots of drama are to be found in the sacred; in ritual, religious observance, and spiritual renewal. The co-option of Alife in the production of technoetic theatre may return drama to its roots. In its origin, it seems likely that sacred drama was not concerned with performance (which would have implied an audience external to the ceremony). Rather its concern was to enact those movements, utterances, gestures (along with colour, light and music) which would maintain or heighten the psychic energy, and the social integrity of the group. Enactment was of a kind of social and psychic utility that performance alone could not emulate.

Recent research in this context has taken me to the Xingu River region of Amazonian jungle where I spent a short period living with the Kuikúro people. Like many groups in this area, the pagé (shaman) has a defining role in daily life. All the activity of the pagé, and of those who interact with him in image making, dancing, chanting, making music, is performative but *is not intended as a public performance*. It is never played to an audience, actual or implicit. No one is watching or is expected to watch what is being enacted. This is not a public performance but a spiritual *enactment.* which entails the structuring or re-structuring of psychic forces. To paint the body elaborately, to stamp the ground repeatedly, to shake the rattle, to beat the drum, to circle round, pace back and forth in unison, is to invoke these forces, to conjure hidden energies. This is an enactment of psychic power not a public performance or cultural entertainment.

So the perspective of sacred theatre, although seen to be at a great distance from our current hypermediated culture, may be of value in viewing the function of works of interactive drama, thereby avoiding the double observer, the phantom audience. The double observer is he who passively, at a distance, watches the observer as interactive agent immersed in dramatic emergence, but whose virtual voyeurism renders the consequences of interactivity void. It reduces interactive dramatic enactment to mere performance; drama-in-a-box or distanced by another kind of proscenium arch, thereby negating the very value of intensive connectivity and immersion that the new dramatic form might have been expected to promote.

Drama as an enactment of mind implies an intimate level of human interaction within the system which constitutes the work of theatre, a theatre without audience in its inactive mode. Eschewing the passive voyeur (the traditional drama viewer) technoetic theatre speaks paradoxically to a kind of widespread intimacy, closeness on the planetary scale. It is the question of intimacy in the relationship between the individual and cyberspace, which must be at the heart of any research into technologically assisted constructions of reality. The quality of intimacy is not strained; it falleth as the gentle rain from heaven....

If we take the canonical, five-fold path that the multimedia artist takes in cyberspace (connectivity, immersion, interaction, transformation, emergence), we can see that it applies also to the development of interactive drama, to technoetic theatre. All the attributes of interactive drama will lead to this overarching quality of emergence. Eventually, the principle of emergence will inform scenario, plot, narrative, audience, in an integrated process. Moreover these qualities will be multiplied, extended and distributed, through the Net, not only in space and location but also in time and duration. The distinction between audience and actor, writer and responder, viewer and viewed with become blurred.

An even greater consequence, will be the merging of cyberspace with the space of the mind.

A truly telematic technoetic theatre, such as seems to be an inevitable outcome of current trends in the technology of distributed intelligence, would put the individual mind on line, in a state of heightened connectivity.

It could be said that technoetic theatre aspires ultimately to the condition of total immersion, providing a radically immersive connectivity within an extended telematic domain. Thus the momentum in theatre to move progressively from the flat screen and proscenium arch, to theatre-in-the-round and the IMAX dome, towards theatre in the Net (or the Net *as* theatre). One consequence of such a development, in which action is widely distributed and ideas, events and narrative sequences emerge through associative links, will be that the sense of time is not only distorted but effectively erased. In netspace, the economy of connectivity replaces the economy of space and time.

The role of dramatist is also to be redefined. Content and meaning are created largely out of the audience's interactions and negotiations with conditions initially set by the "author" but subject to transformation as all contexts are, relative to the fluctuations of content. Technoetic theatre is typically unstable and shifting, constantly in flux; a theatre which parallels life, not through representation or narrative, but in its processes of emergence, complexity, uncertainty and transformation. It finds its place in an emerging culture, which favours the ontology of becoming, rather than the assertion of being, moving towards a post-biological re-materialisation. Technoetic theatre is the theatre of enactment, played without audience, in a network of interacting minds.

Revision is also required of the role of producer, who must now be expected to bring together not simply a 'cast' of live actors, artificial agents or avatars but a dynamic environment, which in turn is shaped by their behaviour and by audience intervention. This means that technoetic theatre must inhabit a complex space: part virtual, part noetic, part material, and always telematic since connectivity is the relational force that drives the whole emergent process. So the discourse of interactive media arts, culled in part from film theory and in part from computer science, must prepare itself for a complete reversal of focus from the much celebrated and much hyped "immateriality" of culture to a radical re-materialisation embodied largely in Alife, and inevitable in its moist manifestation. Such a process of re-materialisation looks to a union of robotics, Alife and telematics as the space of innovation and creative thought.

If there is little evidence of real benefits from this union in current practice it may be that more emphasis has been put upon the demonstration of technological wizardry than on the provision of new and profound experience of the post-biological condition. This is a familiar stage in the evolution of telematic art and interactive media in general. Demonstration takes precedence over experience. Special effects supersede meaning. The material overcomes the spiritual. It is not unlike the situation in science, where description is confused with explanation.

Conclusion

In technoetic theatre, not only is the proscenium arch made redundant but the spacial limitation of the stage per se is eradicated in favour of a limitless network, the horizonless expanse of cyberspace. So equally, the duration of the traditional drama presentation, be it on the live stage, in the cinema, with computer games or on TV, is put aside in favour of the indeterminacy of telematic time which is experientially a sense of no time at all.

Just as Alife will transform our understanding of theatre, so technoetic theatre will transform our understanding of how we can re-enact our identity, re-define ourselves and re-construct our reality, thereby rejecting definitively the inducement to perform roles within a scripted social framework.

The inadequacy of performance and interpretation, in the context of psychic empowerment, self-construction and reality building that technoetic theatre could provide, is contrasted to the efficacy of enactment as shown in many ancient, non-western cultures.

Finally, the fusion of Alife and theatre, should its proponents succeed in realising its interactive, self-generative and globally connective potential, will bring to cyberspace a complexity and variety of meaning and experience of which traditional dramaturgy could only dream.

©Roy Ascott 1998

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Sociality of Robots in Symbiotic Relations with Humans

Tetsuo Ono Michita Imai Tameyuki Etani
ATR Media Integration & Communications Research Laboratories
2-2 Hikaridai, Seikacho, Sorakugun, Kyoto, 619-0288, JAPAN
E-mail: {tono, michita, etani}@mic.atr.co.jp

Abstract

In this paper, we investigate the sociality of robots living together with humans, focusing on the following three points. First, in order to construct a relationship between humans and robots, we propose an interface model with a lifelike agent. In our model, the agent can migrate from a user's mobile PC to a robot when the agent has to work in the physical world. As a result of the migration, a robot can inherit the context from the interactions between a user and an agent so that a relationship between a user and a robot is formed. We show that a robot can guide a user to a destination using this relationship. Second, we propose an agent model using artificial life technology, where each agent pursues a sense of consistency and interacts spontaneously with other robots and an environment. We show that the robots with our model autonomously generate a field of communications between themselves, and furthermore, they promote matchmaking among the guided users in the field. Third, we discuss whether the character and behaviors of humans change during role-playing in the human-robot society, and we compare the results to those of social psychology.

1 Introduction

The *Symbiosis of humans and robots* will be an important theme in the future. This is because robots can provide the physical support that humans need, e.g., carrying things, rescuing people, guiding people alongside, as well as giving the information support that a mobile PC can present to us. However, a method for smooth communication between humans and robots has not been established yet. We can point out that one of the reasons for this comes from problems in the traditional communication model.

In the field of engineering, the model of human communication has so far been based on Shannon's model [6], in which information is transmitted and received on a line, changing direction by turns along to a time progression. However, communication between humans cannot be grasped by the model, because human communication is a more dynamic process realized by participants' sense-making subjectively [2]. We, therefore, have to reconsider the model of human communication. Consequently, this will begin to settle the problems of human-robot communication.

The aim of our research is to investigate the sociality of robots living together with humans. For the realization of the research goal, we focus on the following three points. First, we propose an interface model with a lifelike agent in order to reconsider the traditional model of human communication. Second, we propose an agent model using artificial life technology, in order to realize robot autonomy. We show that the robots with these models can guide humans to destinations, and promote matchmaking among humans. Third, we discuss the function of role-playing between humans and robots.

2 Human-Robot Interface

2.1 Itako Robot and Hyoui Mechanism

It is important to learn how to design the beginnings of interactions between humans and robots. In the traditional model of a human-robot interface, the interaction is begun by a sender's message, e.g., a specified sign and a command. However, a task-oriented design like this is unnatural in daily life because humans do not execute commands that a stranger gives us unexpectedly. We instead ought to focus on simultaneity and embodiment in the beginning of the interactions, i.e., the greeting function in human communications.

We propose an interface model with a lifelike agent in order to construct a relationship between humans and robots. First of all, a user interacts with an agent

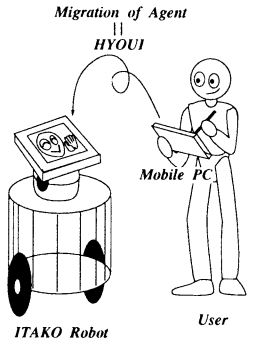


Figure 1: Itako robot and Hyoui mechanism

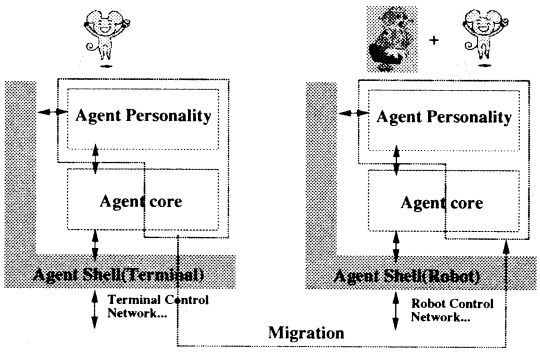


Figure 2: Software structure of the agent

on a mobile PC through the input of a user's interests and the output of an agent's information. In our model, the agent can migrate from the user's mobile PC to a robot when he/she has to work in the physical world (see Figure 1). The migration is called a *Hyoui* mechanism, and the possessed robot is called an *Itako* robot[1]. As a result of the migration, the robot can seamlessly inherit the context from the interaction between the user and the agent so that a relationship between the user and the robot is formed. Notice that this relationship is not formed by the exchange of messages like in the traditional model, but by a function based on simultaneity and naturally embodiment. We expect that a robot can guide a user to a destination using this relationship.

2.2 Implementation

We describe here a method of implementing the Hyoui mechanism. The lifelike agent consists of three layered componets, i.e., an agent personality (AP), an agent core (AC), and an agent shell (AS), which can be rearranged dynamically (see Figure 2). The AP has knowledge-based objects regarding the user and environment, the AC has processing definitions regarding path-planning and presentation strategy, and the AS controls the physical resources of the network and the robot. In the process of the migration, the AP and part of the AC move from the mobile PC to the robot. An unpossessed robot moves autonomously obeying the set initial state.

[1]Hyoui means a process possessed by a soul, and Itako means a soul mediator in the Japanese native religion.

3 Emergence of Communication: Matchmaking

In an information-oriented society, it is important for people to encounter others who have the same interests because of the need to do creative collaborations. We will discuss a communication support environment using on Itako robot as a mediator.

3.1 Autonomy of Agents

The agent on the robot has to have autonomy in order to be regarded as a member of human society and to make sense subjectively in communications. In this section, we focus on a sense of consistency in communications. According to psychological studies, there are many ways to display this feature. For example, a person's heartbeat increases rapidly from fear when he/she crosses a frightening suspension bridge. The rapid heartbeat lasts even after the person has finished crossing the bridge. At that moment, if he/she meets a member of the opposite sex, he/she is likely to be well disposed to that person. The reason is that he/she will have attributed the rapid heartbeat to the other person's attractiveness[2] [5]. In other words, people subjectively generate and maintain a sense of consistency in such a way so that they can adapt to the environment, thereby avoiding the frame problem in artificial intelligence. This is an important factor to agents as well as humans.

3.2 Model Using Artificial-Life Technology

For the purpose of the realization, we propose a communication model in which each agent sponta-

[2]It can, of course, be attributed to other reasons.

neously pursues a sense of consistency in communications with other agents and in interaction with the environment. This sense of consistency means a stable state constructed by dynamical interactions that take place between the internal state, behavior, and environment. Our model of multi-order functions (MOF) uses self-producing systems in artificial life, in particular, Metabolism-Repair Systems ((M, R) systems) [1] in describing the dynamical aspect in the communication process[3].

The MOF model is a hierarchical network structure composed of elements (see Figure 3). An element at a lower level is provided in the following diagram:

$$I_i \xrightarrow{f_i} O_i \xrightarrow{\phi_{f_i}} H(I_i, O_i) \quad (1)$$

Here, I_i is the set of inputs from the environment to the system; O_i is the set of outputs of the system to the environment; f_i is the component of the system represented as a map from I_i to O_i; and ϕ_{f_i} is the repair component of f_i as a map from O_i to $H(I_i, O_i)$, ($H(X, Y)$ is the set of all maps from a set X to a set Y); i identifies each element.

An element at a higher level is described by the following diagram:

$$H(O_i, H(I_i, O_i)) \xrightarrow{F} H(O_i, H(I_i, O_i))$$
$$\xrightarrow{\phi_F} H(H(O_i, H(I_i, O_i)), H(O_i, H(I_i, O_i))) \quad (2)$$

Here, F is a component of the system represented as a map from the set of ϕ_{f_i} to the set of ϕ_{f_i}, and ϕ_F is the repair component of F.

To put it concretely, the elements at the lower levels construct a stable state grounded on sensory-motor coordination. Figure 3 shows a stable state, i.e., the connection among the elements e_4, e_2, and e_3. Simultaneously, the system as a whole becomes stable since the element at the higher level synchronizes with each lower element by interacting with them. Then, the higher element tries to rewrite orders in ϕ_{f_4}, ϕ_{f_2}, and ϕ_{f_3} following its own context. The lower elements also send the current state in ϕ_{f_i} to the higher element.

If a contradiction occurs between the higher and lower levels, it can only be dissolved by interaction between F and ϕ_{f_i} after the fact; however the method is not designed to work before the fact. In other words, the higher level is not in firm control of the bottom levels. The elementary behaviors are fully operative on their own and continue to work even if the higher level is removed. Accordingly, the effect of the higher level is not to select actions directly, but to change the

[3]Refer to [3] for the details of this model.

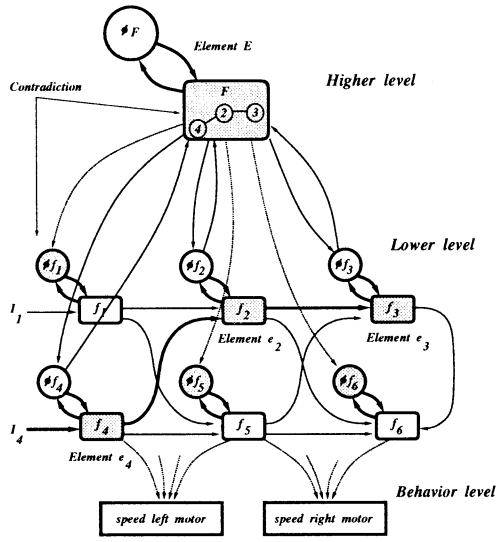


Figure 3: Structure of MOF network

characteristics of the lower levels. The network structure of MOF generates and maintains a sense of consistency autonomously, dissolving any contradiction between the levels by interaction. This effect is grounded on the advantage of the (M, R) systems, which continuously regenerate and achieve the network, producing and maintaining themselves coherently.

The agent with MOF can be expected to autonomously interpret his/her own behavior and environment alone, maintaining the sense of consistency. This mechanism may be beneficial for adaptiveness since it allows qualitative changes in an agent's behavior without changing the supporting mechanisms.

3.3 Matchmaking Mediated by Robots

We can observe an emergence of communication between humans by using robots with the proposed model. For example, we assume that an Itako robot guides a user to a destination in a museum and that the agent on the robot has an internal model of the user's interests and requirements beforehand. If the robot accidentally encounters another robot with the same internal model, the two robots form a field of communication by the entrainment resulting from the characteristics of the proposed model (see Figure 4). Under the influence of this communication, the two users originally having the same interests also begin to communicate with each other. Matchmaking between humans is realized by using the robot as a mediator through the process mentioned above.



Figure 4: Matchmaking through Itako robots

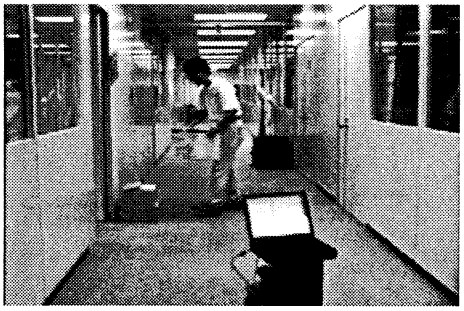


Figure 5: Experiment of role-playing

4 Role-Playing between Humans and Robots

In this section, we discuss role-playing between humans and robots, comparing the results with those of social psychology.

Humans mold a *social-self* through the role they play as a member of the society [4]. In social psychology, experiments concerning human's role-playing have been carried out previously. Zimbardo et al. showed that subjects given the social roles of a prison guard and a prisoner were excessively immersed in those positions through simulations [7]. We would like to discuss whether the character and behaviors of humans are changed through role-playing in the human-robot society in the same way.

In our research, we did experiments in the case in which robots gave commands to subjects (see Figure 5). We then used a post-test to investigate changes in subjects' attitudes and their impressions of the robots. In the early results of the experiments, the subjects had a tendency to dislike robots' commands more than humans' commands. In the future we have to make the factors concerned clear through other experiments. In addition we will study the social relationships between humans and robots, the new comers to human society.

5 Discussion and Conclusions

In this paper, we have investigated the sociality of robots living together with humans, focusing on the following three points, i.e., the robot guidance with our interface model, the matchmaking mediated by the robots with the MOF model, and the role-playing between humans and robots. These points of view shed a new light on communications between humans and robots. That is, we focus on subjective sense-making, simultaneity, and embodiment in communication, which a traditional model like Shannon's [6] cannot grasp. We claim that communication is based essentially on these factors.

Research on robots is important in the field of artificial intelligence and artificial life. This is because such research logically moves our concerns from toy problems to real world problems, and also suggests a useful view to engineering applications. Moreover, in considering *intelligence* and *life* constructively, this research is indispensable to integrating various functions into one material body.

In the days ahead, we will come to live together symbiotically with robots who replace human caretakers and rescue us out of necessity. In this situation, it is meaningful to study the influence of social interactions like role-playing between humans and robots.

References

[1] Casti, J.L. (1988): The Theory of Metabolism-Repair Systems. *Applied Mathematics and Computation*, Vol. 28, pp. 113-154.

[2] Fukaya, M. and Tanaka, S. (1996): *A Theory of Sense Making.* Kinokuniya Syoten. (in Japanse)

[3] Ono, T. and Okada, M. (1998): Consistency Generation Dependent on Situation – Can Robots be in Love? –. *IEEE International Workshop on Robot and Human Communication.*

[4] Saito, I. (eds.) (1987): *The Handbook of Experimental Social Psychology 3: Interpersonal Communication and Attitude Change.* Seishin Shobo. (in Japanese)

[5] Schachter, S. and Singer, J.E. (1962): Cognitive, Social, and Physiological Determinants of Emotional State. *Psychological Review*, 69, pp. 379–399.

[6] Shannon, C.E. (1948): A Mathematical Theory of Communication. *The Bell System Technical Journal*, Vol.27, No.3, pp.379-423.

[7] Zimbardo, P.G., Haney, C., Banks, W.C. and Jaffe, D. (1977): The Psychology of Imprisonment: Privation, Power and Pathology. In J.C. Brigham and L.S. Wrightman (eds.) *Contemporary Issues in Social Psychology, 3rd ed.*, Cole Publishing Company.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Creation of Subjective Value through Physical Interaction between Human and Machine

Takanori Shibata and Kazuo Tanie
Mechanical Engineering Laboratory, Ministry of International Trade and Industry
1-2 Namiki, Tsukuba 305, Japan E-mail: shibata@mel.go.jp

Abstract: Recent advances in robotics have been applied to automation in industrial manufacturing, with the primary purpose of optimizing practical systems in terms of such objective measures as accuracy, speed, and cost. This paper introduces research on artificial emotional creatures that seeks to explore a different direction that is not so rigidly dependent on such objective measures. The goal of this research is to explore a new area in robotics, with an emphasis on human-robot interaction. There is a large body of evidence that shows the importance of the interaction between humans and animals such as pets. We have been building pet robots, as artificial emotional creatures, with the subjective appearance of behaviors that are dependent on internal states as well as external stimuli from both the physical environment and human beings. The pet robots have multi-modal sensory system, actuators, and bodies with artificial skin for physical interaction with human beings.

Key Words: Human-Machine Interaction, Physical Interaction, Pet Robot, Subjective Value, Emergent Emotion

1. Introduction

A human understands people or objects through interaction. The more and longer they interact, the deeper the human understands the other. Long interaction can result in attachment and desire for further interactions. It may also result in boredom. Interaction stimulates humans, and generates motivations for behaviors. There can be cases in which behaviors are not rational.

Objects with which humans interact include natural objects, animals and artifacts. Studies on interaction between human beings and animals show positive effects on psychology, development of children, and so on [1]. Artifacts that affect people in mentally can be called "aesthetic objects". Such effects are subjective and could not be measured simply in terms of objective measures such as accuracy, energy and time.

Machines are also artifacts. Different from the aesthetic objects, machines have been designed and developed as tools for human beings while being evaluated in terms of objective measures [2]. Machines are passive basically because human beings give them goals. Machines will not be active as long as they are tools for human beings.

However, if a machine were able to generate its motivation and behave voluntarily, it would have much influence to an interacting human. At the same time, the machine should not be a simple tool for humans nor be evaluated only in terms of objective measures. Subjective evaluation is important. For a human, multi-modal stimulation should be influential. People interacting with the machine or observing the interaction may consider the machine as an artificial creature. Behaviors of the machine may be interpreted as emotional.

There are many studies on human-machine interaction. Here, we don't discuss studies on human factors in controlling machines used as tools. In other studies, machines recognize human gestures or emotions by sensory information, and then act or provide some information to the human. However, modeling gestures or emotions is very difficult because these depend on the situation, context and cultural background of each person.

Concerning action by a machine toward a human, an artificial creature in cyber space can give only visual and auditory information to a human. A machine with a physical body is more influential on human mind than a virtual creature.

Considerable research on autonomous robots has been carried out. Their purposes are various such as navigation, exploration and delivery in structured or unstructured environments while the robots adapt to the environments. Also, some robots have been developed to show some emotional expressions by face or gestures. However, even though such robots have physical bodies, most of them are not intended to interact physically with a human.

We have been building pet robots as examples of artificial emotional creatures [1, 3, 4]. The pet robots have physical bodies and behave actively while generating motivations by themselves. They interact with human beings physically. When we engage physically with a pet robot, it stimulates our affection. Then we have positive emotions such as happiness and love or negative emotions such as anger, sadness and fear. Through physical interaction, we develop attachment to the pet robot while evaluating it as intelligent or stupid by our subjective measures.

The chapter 2 discusses subjectivity and objectivity. The chapter 3 discusses emergence of emotional behaviors through physical interaction. The chapter 4 introduces newly developed pet robots [5]. Finally, the chapter 5 concludes this paper.

2. Objectivity and Subjectivity

Science and technologies have been developed through objectivism. Because of this, people can share and use their scientific and technological knowledge in common. When we design machines, we need to use such objective knowledge. A machine which has high value evaluated in terms of objective measures such as speed, accuracy, and cost is useful as a tool for human beings, especially for automation.

A machine that interacts with a human is not always evaluated by such objective measures. People evaluate a machine subjectively. Even if some machines were useless in terms of objective evaluation, some people put high subjective value on them. Such machines could be considered as aesthetic objects.

When we design robots that interact with human beings, we have to consider how people think of the robots subjectively. This paper deals with pet robots to investigate subjectivity for designing robots friendly to human beings.

3. Emergence of Emotional Behavior through Physical Interaction

There is an enormous number of studies on emotions. Also, we can make models of emotions by observing many people. However, we can not say which model is correct or even the best. We have many words to express our own emotions, but we don't have the same definition of internal states of our bodies. Emotions are evoked in some situation, and depend on context and cultural background. Therefore, it is difficult to establish a general model of emotions. For example, if a subject and another person are interacting, the person's interpretation of emotions of the subject is not always the same as that of the subject himself. Even if they had long relationship, they would interpret the emotions in different ways.

We are taking a position that emotions emerge through interaction with the environment as Picard classified research on emotions [1, 3, 4, 6]. There is some research on emergent emotions. Toda emphasized the importance of studying whole systems including perception, action, memory, and learning [7]. He proposed a scenario with a fungus eater to illustrate how emotions would emerge in a system with limited resources operating in a complex and unpredictable environment. Toda's robot has the goal of collecting as much uranium ore as possible, while regulating its energy supply for survival. The robot has rudimentary perceptual, planning, and decision-making abilities. Toda proposed urges that are motivational subroutines linking cognition to action, and argued the robot would be emotional with the urges. The urges are triggered in relevant situations and subsequently influence cognitive processes, attention, and bodily arousal. An observer of the robot would interpret that the robot's behaviors are emotional. Pfeifer implemented urges in a mobile robot and showed emergence of emotional behaviors through interaction between the robot and its environment [8].

Braitenberg explained emergent emotions by means of his simple mobile robot, which had two light sensors and two motors [9]. When the robot sees a light source straight ahead, the robot moves toward it, and bangs into it, hitting it frontally. When the source is not straight ahead, then the robot turns and moves so that it still approaches the source and hits it. An observer could interpret the robot's behavior as aggressive as if the robot felt a negative emotion toward the light source. When sensors and motors are wired so as to linger near the source and not damage the robot, the behavior could be interpreted as a more favorable emotion. The robot could have different behaviors in the same environment when its internal system is changed. These mean that emotions have emergent properties and depend on interaction with the environment.

Brooks argued that situatedness, embodiment, intelligence and emergence are key ideas of behavior-based robots [10]. The key idea of situatedness is that the world is its own best model. The key idea of embodiment is that the world grounds the regress of meaning-giving. The key idea of intelligence is that intelligence is determined by the dynamics of interaction with the world. The key idea of emergence is that intelligence is in the eyes of the observer.

When a human and a pet robot interact with each other, they stimulate and affect each other. We call this 'coupling' [4]. When the human evaluates the robot, the human is an observer and a subject at the same time. Following Brooks' ideas, the intelligence of the robot could be determined by the dynamics of interaction with the subject and environment. Also, the subject would interpret or measure intelligence of the robot with his own eyes. The subject's interpretation depends on his knowledge and experiences related to the robot and its designer. Therefore, the robot's intelligence depends on the subject's intelligence.

At this point, we don't have an explicit definition of intelligence. However, as Minsky suggested, we doubt whether machines can be intelligent without emotions (which doesn't mean emotion models but emotional appearance of behavior in observer's view) [11]. Therefore, we consider that emergent emotions are key for intelligence.

As we design pet robots as artificial emotional creatures, an interacting human does not give them goals nor tasks. Pet robots are allowed to generate their own goals and motivations for survival in the world. Therefore, contrary to Asimov's "The Three Laws of Robotics" [12],

1) They would protect themselves.
2) They would not obey human beings.
3) They would injure human beings.

These allowances are the key to let people interpret that pet robots are like living creatures.

4. Pet Robot as Artificial Emotional Creature

4.1 Previous Research

We investigated subjective interpretation of robot's behaviors in psychological experiments, in which a picture of a dog was equipped with a 1 DOF tail and subjects were asked to interpret emotions of the dog by wagging tail [4]. Then, a simple tactile sensor was added to the system and the tail wagged depending on stroking the tactile sensor by subjects. In the first experiment, subjects interpreted meaning of wagging by visual and auditory information. In the second one, subjects had tactile information as well as vision and audition. Interpretations of emotions were various because of knowledge of dogs; for example, some had experience of owning dogs. However, the second experiment was much more impressive for most subjects because of physical interaction with tangibility.

With this result, we developed a pet robot that had visual, auditory, and tactile sensors, a tail with 1 DOF, and mobility by three wheels [4]. We emphasized "tangibility" for physical interaction between a human and a robot, different from other research [13]. For this purpose, we developed a new tactile sensor that consisted of a pressure sensor and a balloon covered with synthetic skin. The robot could sense pushing, stroking, and patting. It behaved depending on its internal state that consisted of current input from sensors and regressive input from itself. A human interacting with the robot by touching or stroking got visual, auditory and tactile information. The human felt softness and nice texture like real creatures. Depending on the information, the human changed his behavior. This loop was considered as coupling between the human and robot. Even though the robot didn't have explicit emotion model, people interacting with the robot interpreted the robot's behaviors were emotional.

From viewpoints of physical interaction and subjectivity, there were deficits that the robot was too heavy to hug or hold, and didn't look like real animals in its appearance. Therefore, we developed two pet robots: one is a seal robot (Fig. 1), and the other is a cat robot (Fig. 2) that has more complex structure and more functions.

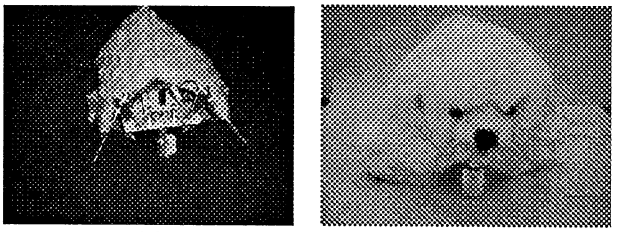
4.2 Small and Animal Like Pet Robots

4.2.1 Seal Robot

A seal robot has a simple structure in order to investigate emergent emotions through physical interaction. The robot has two legs with two servomotors. Each leg has a clutch bearing at a contacting point with floor. At front and back of its body, the robot has two supports. The front support is a caster. The back support has a clutch bearing at the contacting point with floor. There are four contacting points in total and three have clutch bearings in the same direction. When the robot moves two legs back and forth at the same time, it moves forward like crawling. When it moves two legs back and forth alternately, it doesn't move forward but it shakes its body. When one leg was fixed at back and the other leg moves back and forth, the robot turn to a direction of the fixed leg's side. Concerning on sensory system, the robot has two whiskers which sense contact with its environment, and two pressure sensors with balloons which sense pushing, patting and stroking on its body. The robot has 6811 CPU inside to control itself. Its weight is about 1.0 [kg]. We assumed those, as innate characteristics, the robot likes stable pressure on its body and dislikes being touched its whiskers.

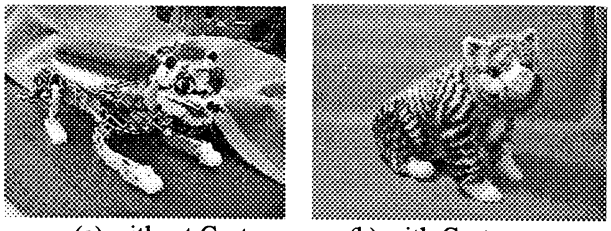
The robot has internal state depending on sensory information and recurrent information. A neural oscillator with two neurons generates motion of each leg, and the internal state was input to a neuron. Phase of the two oscillators was controlled by sensory information.

The robot had some rules to generate its motivation, to change its attention, and to control movement of legs, but it didn't have explicit model of emotions. When people interacted with the robot, they interpreted the robot's behavior differently with some words of emotion to express what the robot was doing. As the movement of the robot depended on context, internal state made robot's behavior more complex in interpretation than the number of given functions. This was the effect of emergent emotions. Complexity of interpretation depended on subjects because it was very subjective view.



(a) without Costume (b) with Costume

Fig. 1 Seal Robot



(a) without Costume (b) with Costume

Fig. 2 Cat Robot

4.2.2 Cat Robot

Cat robot has more complex structure than the seal robot. It was built by OMRON Corp. [5] The robot has tactile, auditory, and postural sensors to perceive human action and its environment. For tactile sensors, it has four piezo-electric sensors on its back and one on its head, and one micro switch at its chin and two at its cheek. The robot can recognize stroking, hitting and touching. For audition, the robot has three microphones for sound localization and one for speech recognition. For posture, the robot has acceleration sensors for three axes. Information is processed one CPU with two DSPs for sensors and one IC for speech recognition. Its weight is 1.5 [kg] including battery.

For movement, the cat robot has one actuator of eyelid, two for neck, two for each front leg, and one for a tail. In total, it has eight actuators. There are one passive joint at front leg's ankle and three passive joints at each rear legs.

As the robot has complex structure, an emotion model is implemented. Therefore, the robot has some patterns and states of emotion for emotional expression, and some model to recognize its environment and human's action. For example, when the cat robot is called by name, it recognizes the word, detect direction of sound source, turn its face to the direction, blink its eyes for eye contact with a human. Also, when robot's internal state is happy, it cries while moving actively. However, in order to keep emergent property in emotional behavior through physical interaction, motion patterns, such as trajectory of a leg, are generated depending on internal states of the robot. Therefore, it is difficult for a subject to predict robot's action.

4.3 Discussions

Each robot has a costume of seal or cat. Without a costume, almost nobody would associate a seal with Fig. 6 (a). A costume gives quite various associations to subjects even without dynamic interaction. For example, in the case of cat robot, people who have experience owing a cat, expect the robot to react in the same way as a real cat. Therefore, when they interpret the robot's behavior, they always compare it with that of real cat.

At this point, we are evaluating the system by statistical data of subjects' impression. As the subjective data is varied, it is difficult to deduce out general principles for designing pet robots with high subjective value. We have to investigate a method to analyze subjective evaluation. A way to evaluate how much people liked a pet robot would be introducing the robot in a commercial market, and counting the number of sales.

5. Conclusions

We developed pet robots as artificial emotional creatures that generated their motivation and behavior through physical interaction with human beings by means of multi-modal information, especially tactile information. These robots have been developed to investigate emotional behaviors emerging through physical interaction with human beings.

We discussed subjectivity in interpretation and evaluation of robot's behavior in physical interaction. Evaluation methods will be investigated with more psychological experiments in the future.

Acknowledgment

This research had been done partially at the AI Lab., MIT. The authors appreciate to Prof. R. Brooks. We also appreciate to Prof. R. Pfeifer, the AI Lab., Univ. of Zurich, for discussion on emergence of emotion.

References

1. T. Shibata, et al., Emotional Robot for Intelligent System - Artificial Emotional Creature Project, Proc. of 5th IEEE Int'l Workshop on ROMAN, pp. 466-471 (1996)
2. H. Petroski, Invention by Design, Harvard University Press (1996)
3. T. Shibata and R. Irie, Artificial Emotional Creature for Human-Robot Interaction - A New Direction for Intelligent System, Proc. of the IEEE/ASME Int'l Conf. on AIM'97 (Jun. 1997) paper number 47 and 6 pages in CD-ROM Proc.
4. T. Shibata, et al., Artificial Emotional Creature for Human-Machine Interaction, Proc. of the IEEE Int'l Conf. on SMC, pp. 2269-2274 (1997)
5. T. Tashima, S. Saito, M. Osumi, T. Kudo and T. Shibata, Interactive Pet Robot with Emotion Model, Proc. of the 16th Annual Conf. of the RSJ, Vol. 1, pp. 11, 12 (1998)
6. R. Picard, Affective Computing, The MIT Press (1997)
7. M. Toda, Design of a Fungus-Eater, Behavioral Science, 7, pp. 164-183 (1962)
8. R. Pfeifer, 'Fungus Eater Approach' to emotion: A View from Artificial Intelligence, Cognitive Studies, The Japanese Society for Cognitive Science, 1 (2), pp. 42-57 (1994)
9. V. Britenberg, Vehicles: Experiments in Synthetic Psychology, The MIT Press (1984)
10. R. Brooks, Intelligence without Representation, Mind Design II (John Haugeland ed.) The MIT Press, pp. 395-420 (1997)
11. M. Minsky, The Society of Mind, Simon & Schuster (1985)
12. I. Asimov, I, Robot, Doubleday (1950)
13. M. Fujita and K. Kageyama, An Open Architecture for Robot Entertainment, Proc. of Agent'97 (1997)

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Task Sharing in a Multiple Robot System Controlled by Macro-Commands of an Operator

Kazuya OHKAWA

University of Tsukuba
1-1-1 Tennodai, Tsukuba
Ibaraki, 305-8573, Japan
okawa@mel.go.jp

Takanori SHIBATA and Kazuo TANIE

Mechanical Engineering Laboratory, MITI
1-2 Namiki, Tsukuba
Ibaraki, 305-8564, Japan
shibata@mel.go.jp, tanie@mel.go.jp

Abstract

This paper proposes a method for controlling task sharing in a multiple robot system depending on situations in order to achieve some tasks. In our study, we assumed that each robot had some programs to achieve all tasks and therefore the robot could change the current task by selecting these programs. Robots may be able to share tasks among robots if each robot selects its own program considering the other robots' selected program. However, the appropriate task sharing expected by an operator may be changed hour by hour, and therefore, the operator can not give rules to share tasks. Each robot should modify its own standard to select the appropriate program based on commands provided by an operator. The operator provides macro-commands for the whole robots, and each robot modifies its own standard of program selection based on the commands. As the result of the modifications, the operator may be able to control task sharing among robots by macro-commands depending on situations. We verify the effectiveness of the proposed method by simulations.

1 Introduction

In the future, we may prepare many robots to achieve a project including several tasks, and make these tasks execute by them, simultaneously. In this case, an operator must control task sharing among robots depending on progress of tasks. The simplest controlling method to share tasks among robots is that an operator determines an appropriate task sharing in advance and gives each robot an exclusive program. In this case, however, the operator must re-determine new task sharing and give new program to each robots whenever the operator desires the different task sharing depending on situations. Another method is that an operator prepares programs to achieve all tasks and equipment for communication with the other robot. Each robot can select an appropriate program considered the other robot through communications. In this case, however, robots may need enough time to communicate with all robots when the number of robots is increased.

We considered these problems and proposed an algorithm[1] that has programs to achieve all tasks. In the algorithm, we assumed that each robot could select one of these programs based on its own standards of program selection, and therefore, each could change tasks depending on situations. And each robot communicates with the neighboring robots. And therefore, we may be able to apply the algorithm when the number of robots is increased. We detail about the algorithm in section 2.

We considered that an operator prepared some robots had the algorithm and made them share appropriately. In this case, the appropriate task sharing may be changed depending on situations, and therefore, the operator can not give rules as standards of program selection in advance. The operator can change these rules depending on situation, however, it is difficult to prepare them when the number of robots is increased. If there are many robots, the operator can not observe each robot individually. The operator may observe a group of robots and provide macro-commands for them. In this paper, therefore, we propose a method for controlling task sharing among robots by macro-commands provided for a group of them. We detail the method in section 3. And we describe a method for providing commands in section 4. In section 5, we show the result of simulations that are performed to confirm the effectiveness of the proposed method. In the final section, we describe conclusions.

2 Algorithm for Cooperation

We proposed the algorithm as shown in Figure 1 in order to change and share tasks among robots. The algorithm was composed of five modules that were *behavior module*, *selection module*, *sensor module*, *evaluation module*, and *frustration module*.

In order to select one of behavioral programs depending on situation, the algorithm must have programs to achieve all tasks in advance. In our study, therefore, we prepared *behavior module*, and provided all programs to achieve tasks in the module. Also, each robot requires having the capability to select one of behaviors in the behavior module. Therefore, we prepared *selection module*. In this case, however, robots may not share tasks appropriately because each robot selects one of behaviors from a selfish motive. In order to share tasks among robots, each robot must select their own behavior considered the other robots' behavior. Therefore, *sensor module* was prepared to obtain information about the relationship with the other robots. In this case, however, robots may not share tasks appropriately in the beginning, because each robot has not learned an appropriate behavioral selection depending on each situation through enough experiments. In order to learn an appropriate selection, *evaluation module* was prepared for each robot. Each robot may select one of behaviors considered the other robots' behaviors, however, there is a possibility that the robot can not execute the current task according to circumstances. Therefore, the *frustration module* is prepared to change the current behavior to another behavior when the robot continues to select the same behavior for a while.

The robot having the algorithm may share tasks considered the other robots' behavior if the evaluation module is provided depending on each situation appropriately.

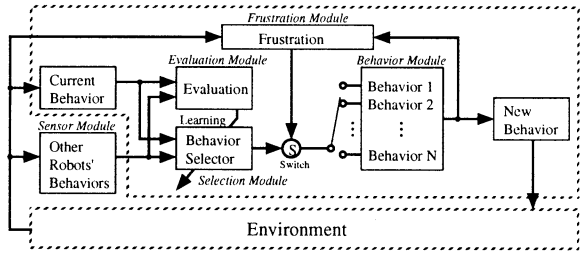


Fig.1: Algorithm for cooperation

3 Modification of Evaluation for Learning

Each robot using the algorithm learns an appropriate behavior selection considered the other robots' behavior. In this case, the robot estimates the result of behavior selection by using *evaluation module*. The evaluation is shown in Table 1. In the table 1, B1 and Bn are Behavior 1 and Behavior n, respectively.

Therefore, the operator may be able to control their behaviors if the operator can change their evaluations depending on each situation. In this paper, we try to change their evaluations by macro-commands that are provided for a group of them. The macro-command was assumed simple command as 'good' and 'bad', because it may be difficult to estimate task sharing among robots in detail.

In order to learn an appropriate behavior selection considered the other robots' behavior, we applied Q-learning that is a kind of reinforcement learning, and therefore an operator must care to change the evaluation. Because robots may not learn the appropriate behavior selection if all values of the evaluation have conditions, such as less than 0, more than 0, 0. These conditions mean penalties only, rewards only and no estimate. In order to evade these conditions, we propose a method that each robot modifies its own evaluation based on macro-commands provided by an operator.

$\mathbf{A}$ and $\mathbf{X}$ are a set of finite discrete of the possible behaviors and the other robots' relationship, respectively. At certain time t, behaviors of self and the other robots' relationship are defined by $a_t (\in \mathbf{A})$ and $\mathbf{x}_t (\in \mathbf{X})$. Each robot stores macro-commands provided by an operator in $\mathbf{R}(a_t, \mathbf{x}_t)$ at each situation. These values of $\mathbf{R}$ at all situations are initialized by 0 at the beginning. And when the operator provides 'good' or 'bad' as a macro-command for a group of robots, $\mathbf{R}(a_t, \mathbf{x}_t)$ is modified +1 or −1, respectively.

Table 1: Behavioral evaluation for learning

Behavioral Evaluation		Other Robots' Relationship		
		(B1, B1, ...)	(B1, B2, ...)	...
Current Behavior	Behavior 1	E (1,1,1, ...)	E (1,1,2, ...)	...
	...	...	...	...
	Behavior n	E (n,1,1, ...)	E (n,1,2, ...)	...

If each robot uses these values for learning, there is a possibility that the robot has evaluations satisfied the foregoing bad conditions. Therefore, each robot modifies its own evaluation $\mathbf{E}(a_t, \mathbf{x}_t)$ using the following equation:

$$\mathbf{E}(a_t, \mathbf{x}_t) = \mathbf{R}(a_t, \mathbf{x}_t) - \frac{\sum_{i \in \mathbf{A}} \sum_{j \in \mathbf{X}} \mathbf{R}(i, j)}{\sum_{i \in \mathbf{A}} \sum_{j \in \mathbf{X}} \|\mathbf{R}(i, j)\|} \tag{1}$$

By using the foregoing equation, all values of the modified evaluation are not less than 0 or more than 0. Therefore, each robot may be able to learn an appropriate behavior selection.

4 Conduct Method for Task Sharing

Each robot modifies its own evaluation based on the foregoing equation whenever an operator provides macro-commands for a group of robots. And each robot learns an appropriate behavior selection considered the other robots' relationship.

A certain robot's behavior and the other robots' relationship are defined by $a_t (\in \mathbf{A})$ and $\mathbf{x}_t (\in \mathbf{X})$ at certain time t, respectively. When the robot selected $a_{t+1} (\in \mathbf{A})$ at time $t+1$. Q-values is modified depending on each robot's evaluation as follows:

$$Q_{new}(a_t, x_t, a_{t+1}) \leftarrow Q_{old}(a_t, x_t, a_{t+1}) + \Delta Q \tag{2}$$

where ΔQ is calculated as the following:

$$\Delta Q = \alpha \left\{ \begin{array}{l} \mathbf{E}(a_{t+1}, \mathbf{x}_{t+1}) \\ \quad + \gamma \max_{b \in \mathbf{A}} Q_{old}(a_{t+1}, \mathbf{x}_{t+1}, b) \\ \quad - Q_{old}(a_t, \mathbf{x}_t, a_{t+1}) \end{array} \right\} \tag{3}$$

where α is learning rate and γ is discounting factor.

Each robot learns an appropriate behavior selection by using the foregoing equations. In this case, however, robots may spend many trials to find a good task sharing in the beginning. Because, robots can not obtain 'good' command until they make a good task sharing by chance. Therefore, we considered that an operator provides a good command in the beginning even if robots do not share tasks exactly. After that, the operator enhances the quality of sharing after robots executed the task sharing for a while. If the operator enhances the quality soon when robots execute good task sharing, they may not be able to continue the task sharing, because each robot requires several trials to learn the sharing. In this paper, we tried to calculate how many trials each robot requires for learning.

We assumed that robots obtained a reward from an operator, and continued to select its behavior. When the Q-value of its behavior is Q_1 at the current time, we can calculate Q_n and Q_{n-1} from the equation (2).

$$\begin{aligned} Q_n &= \sum_{k=1}^{n-1} \alpha \mathbf{E}_t (1 + \alpha\gamma - \alpha)^{k-1} + Q_1 (1 + \alpha\beta - \alpha)^{n-1} \\ Q_{n-1} &= \sum_{k=1}^{n-2} \alpha \mathbf{E}_t (1 + \alpha\gamma - \alpha)^{k-1} + Q_1 (1 + \alpha - \alpha)^{n-2} \end{aligned} \tag{4}$$

where Q_n and Q_{n-1} are Q-values of its behavior after the selection n trials and $n-1$ trials, respectively. $\mathbf{E}_t$ is a evaluation value at the time t.

By using equation (4), we can calculate the additional value using the following equation:

$$\begin{aligned} \Delta Q_{n-1} &= Q_n - Q_{n-1} \\ &= \alpha (1 + \alpha\gamma - \alpha)^{n-2} \{\mathbf{E}_t + Q_1 (\gamma - 1)\} \end{aligned} \tag{5}$$

If the ΔQ_{n-1} is small, it means that its robot's learning was finished. In this case, we assumed that the learning was converged when the ΔQ_{n-1} was less than T. By using these equations, we can calculate the number of trials that is required to learn, as the following:

$$\textit{if } \Delta Q_{n-1} \le T$$

$$n \ge 2 + \frac{\log \left\{ \dfrac{T}{\alpha \{\mathbf{E}_t + Q_1 (\gamma - 1)\}} \right\}}{\log (1 + \alpha\gamma - \alpha)} \tag{6}$$

In general, Q_1 has a value more than 0, because the behavior having its Q-value was an appropriate behavior in its situation. Therefore, we assumed that Q_1 was 0 when we wanted to calculate the value of n.

5 Simulation

We performed simulation in order to confirm the effectiveness of proposed method. At first, we tried that an operator controlled nine robots by using the method of section 3. Next, we tried that the operator conducted a group of robots by using the method of section 4 in order to share tasks more quickly.

5.1 Conditions of Simulations

We applied the algorithm as shown in Figure 1 to nine robots. The algorithm has three behavioral programs to achieve three tasks. Each robot can achieve one of tasks by selecting one of these programs. In this simulation, we assumed that each robot could obtain the other robots' selected task by its own sensors.

The purpose of this simulation is that an operator controls nine robots in order to share tasks. We assumed that the operator changes targets of task sharing as Table 2 depending on the number of trials. The operator observes a group of robots and provides macro-commands such as 'good' or 'bad' for them. Each robot modifies its own evaluation based on the macro-commands by using the equation (1). All values of evaluation were set to 0. And initial values used in the learning function were chosen as random parameters.

5.2 Results of Simulations

We executed this simulation in the foregoing conditions. The result of the simulation is shown as Figure 2. In the figure, the horizontal axis means the number of experimental trials, and the vertical axis means the number of robots selected each task. In this simulation, we confirmed that an operator could control nine robots to desired task sharing by macro-commands.

In order to share tasks by robots more quickly, we tried that the operator conducted a group of robots by using the method of section 4. In this simulation, the operator enhanced the quality of task sharing gradually after these robots continued the sharing for 17 trials. The other conditions were set by the conditions of the foregoing simulation. The result of this simulation is shown as Figure 3. In the simulation, we confirmed that the operator could control task sharing among robots by commands more quickly by comparing to Figure 2.

6 Conclusion

We proposed a method for controlling task sharing among robots by macro-commands that are provided for a group of robots from an operator. The operator gives an algorithm as shown Figure 1 to each robot. And the operator observes the group of robots, provides macro-commands, such as 'good' or 'bad'. Each robot obtains these commands and modifies its own evaluations that are used for learning an appropriate behavior selection depending on each situation. As the result of the modifications, robots can share tasks based on macro-commands provided by the operator. We verified the effectiveness of the proposed method by simulations.

Reference

1. Shibata T, Ohkawa K and Tanie K, Spontaneous Coordinated Behavior of Robots through Reinforcement Learning. ICNN'95, pp.2908-2911 (1995)

Table 2: Targets of task sharing

Trials	Task 1	Task 2	Task 3
1 – 1000	3 robots	3 robots	3 robots
1001 – 2000	1 robot	1 robot	7 robots
2001 – 3000	5 robots	4 robots	non
3001 – 4000	2 robots	3 robots	4 robots

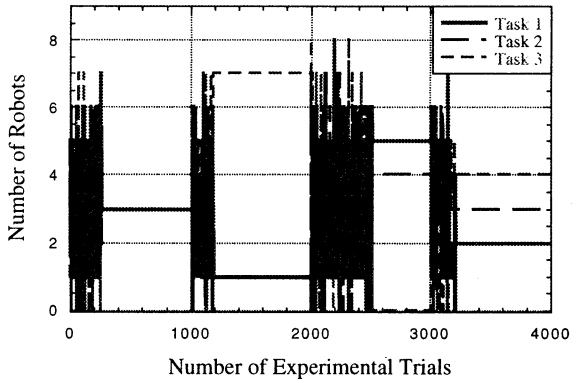


Fig.2: Task sharing based on macro-commands

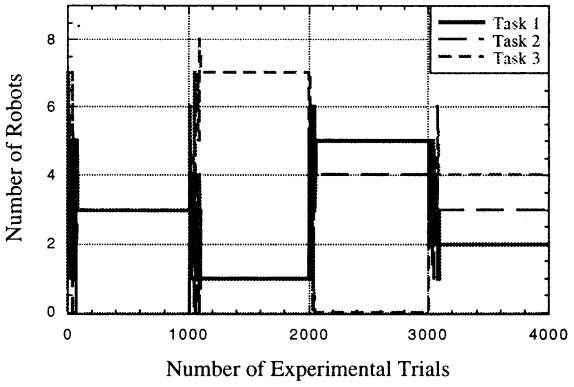


Fig.3: Conduction of task sharing among robots

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Evolutionary Parallel Collaborative Computation on Intelligent Agents

Toru Yamaguchi, Naoki Kohata, Makoto Takahide, Takanobu Baba and Hideki Hashimoto*

Department of Information Science, Faculty of Engineering, Utsunomiya University
7-1-2, Youtou, Utsunomiya 321-8585, Japan
**Institute of Industrial Science, University of Tokyo*
7-22-1 Roppongi, Minato-ku, Tokyo 106-8558, Japan

kohata@sophy.is.utsunomiya-u.ac.jp

Abstract

This paper proposes an evolutionary parallel collaborative computation instead of conventional GA (Genetic Algorithm), in order to realize such intelligent agents as welfare robots which assist humans. This evolutionary computation is realized by applying chaotic retrieval, Soft DNA(Soft computing oriented Data driven fuNctional scheduling Architecture) and collaboration among agents. Plural agents perform evolution at each own situation separately, and then all knowledge items of the agents are integrated collaboratively, like collaborative recommendation method in AI. We show some applications of this evolutionary computation, i.e., multi-agent robots which move abreast, ITS(Intelligent Transport System) and so on. Essentially, the process of this evolutionary computation is parallel processing. Therefore, we implement its parallel processing algorithm on A-NET (Actors NETwork) parallel object-oriented computer, and show the usefulness of parallel processing for proposed evolutionary computation.

Keywords: Evolutionary computation, Chaotic retrieval, Associative memories, Parallel processing

1. Introduction

Recently, evolutionary computation models on Alife (Artificial life) have been researched by computer[1]. Nowadays, its typical approach method is GA (Genetic Algorithm). Conventional GA is an algorithm based on traditional Darwinism. On the other hand, in recent years, new theories of evolution except Darwinism have been advocated. Nakahara et al. have advocated virus theory of evolution, and explain rapid evolution which can not be explained by mutation and natural selection [2]. Yomo et al. have advocated evolution based on competitive coexistence, and argue that evolution is not simple optimization because of interaction among life [3]. In any case, it is certain that evolution of actual life is not such simple processes as conventional GA. Above all, we think there are not only genetic factors but also other factors (e.g., cultural factor) in evolutionary process of brain or its intelligence. In addition, it is said that evolution is irreversible process which does not enable the life to become again the exactly same life as it used to be. It seems to us that there is chaos in this complexity of evolution.

Therefore, we propose evolutionary computation of intelligence by chaotic dynamics and Soft DNA (Soft computing oriented Data driven fuNctional scheduling Architecture) and collaboration among agents as shown in **Fig. 1**[4]. We explain this evolutionary computation and Soft DNA in the following chapter.

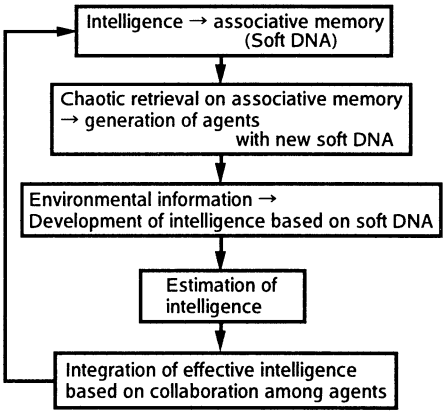


Fig. 1 Chaotic evolutionary collaborative computation of intelligence using soft DNA

On the one hand, in the society which is filled with old people, the welfare agent robots which assist the old or the sick people are requested as shown in **Fig. 2**. The welfare robots have to move in a suitable formation, in cooperation with other agents, humans and the outer environment. We have to acquire the knowledge of cooperative work efficiently. Therefore, we apply the proposed evolutionary computation to the multi-agent robots which move abreast as an example of cooperative work in welfare robots and we also apply this evolutionary computation to ITS(Intelligent Transport System). The brain of this welfare robot is constructed by the intuition-based agent model as shown in **Fig. 2**. This agent model consists of hierarchical fuzzy knowledge that uses associative memories[5], and it imitates human creativities in order to adapt itself to its environmental changes, conceiving new ideas based on current knowledge by chaotic retrieval [10]. Each hierarchical part retrieves the knowledge based on fuzzy associative inference on associative memories.

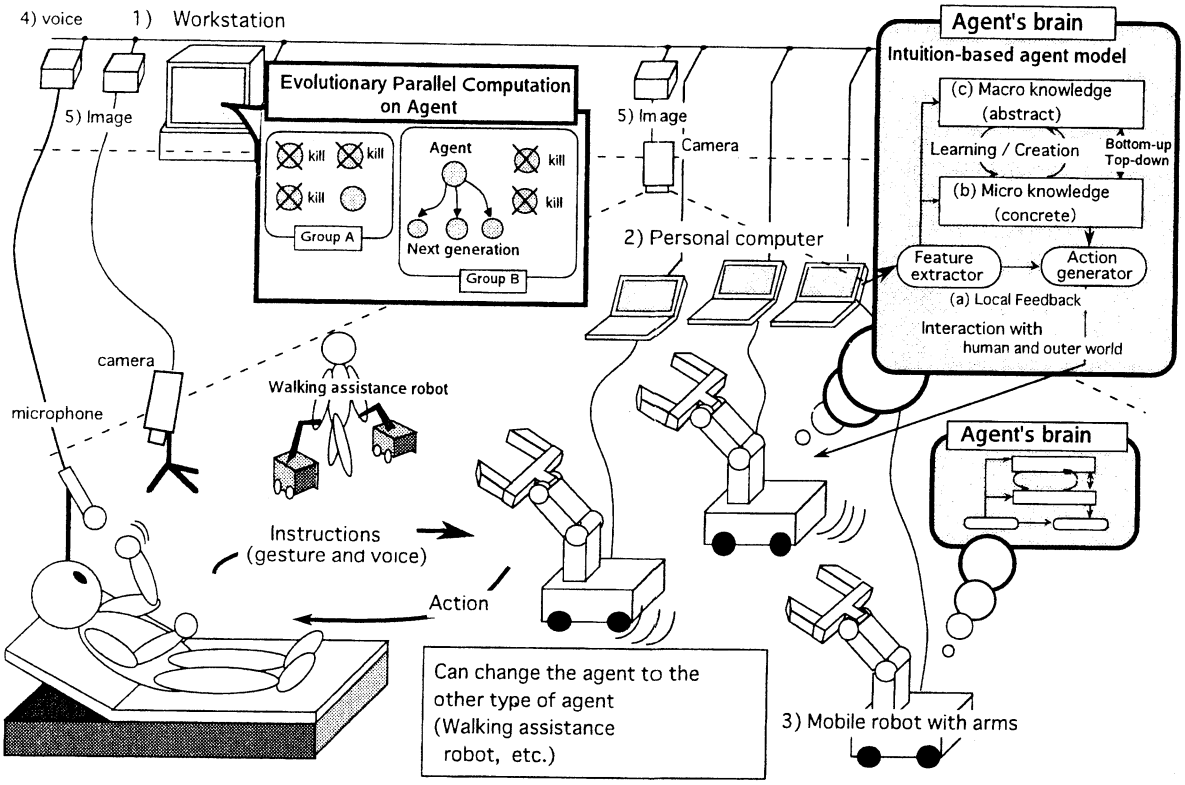


Fig. 2 Welfare intelligent agents and intuition-based agent model

Essentially, this inference in each part is parallel processing, and these hierarchical parts also work in parallel. Furthermore, a large number of agents work in parallel on a multi-agent model and its evolutionary computation. Therefore, we undertook parallel processing according to these parallel properties in the brain and in nature. We implement a parallel processing algorithm on A-NET(Actors NETwork) parallel object-oriented computer[8], and show its usefulness.

2. Soft DNA and its Evolutionary Computation

2.1. Soft DNA

We propose a new method for development of intelligence in order to realize dynamic cooperation in the intelligent agents. We call this new method "Soft DNA (Soft computing oriented Data driven fuNctional scheduling Architecture)". Soft DNA aims to imitate the idea of the developmental process, such as the body plans in actual life based on biological DNA(DeoxyriboNucleic Acid). **Fig. 3** shows the image of soft DNA compared with biological DNA. In biological DNA, the genes called "Homeo box genes" dynamically control the body development of an individual in actual life based on the concentrations of proteins in cells. The control architecture called "soft DNA" dynamically controls the development of intelligence in each agent based on environmental information, in order to achieve dynamic cooperation. Biological DNA has sets of genes that are related to each body part such as the head, chest, abdomen, and tail. These sets of genes are each called a homeo box. Similarly, soft DNA has boxes of intelligence (made by soft computing, i.e., associative memories, neural networks, fuzzy logic, chaos, and so on) that are related to various environments, and a suitable box of intelligence is developed according to the environmental information.

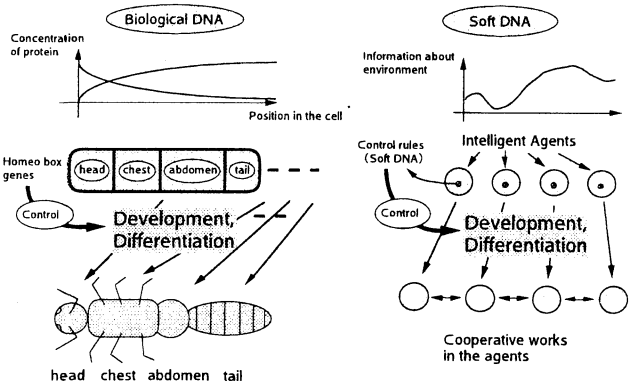


Fig. 3 Image of soft DNA

2.2. Evolutionary Computation on Soft DNA

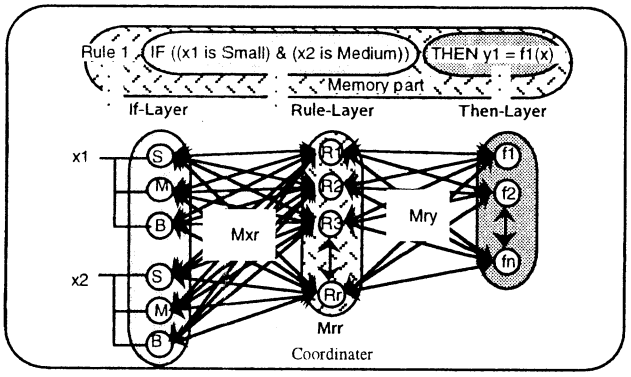


Fig. 4 Fuzzy associative memory organizing units system

The proposed soft DNA consists of some boxes which are made by associative memory system named FAMOUS(Fuzzy Associative Memory Organizing Units System)[5]. We use CFAMOUS(Chaotic FAMOUS) to carry out evolutionary computation on soft DNA. We simulate the proposed evolutionary computation in parallel on A-NET parallel computer which is explained in the next chapter.

FAMOUS (**Fig. 4**) represents fuzzy knowledge using several BAMs (Bi-directional Associative Memories)[7]. It carries out fuzzy associative inference. An initial input is also continuously added as an external input to the if-layer. The rule-layer has a self-connection called the coordinator. This excites one neuron and inhibits others. If the correspondence between the rule-layer and the then-layer is one-to-one, one of these layers can be omitted. This system performs associative inference that causes an input pattern to approach the nearest pattern using top-down and bottom-up processing (i.e., network reverberation). This propagates the activation values of each node. This fuzzy associative inference is essentially parallel processing. CFAMOUS applies the chaotic retrieval to the retrieval process of the memorized patterns in FAMOUS. The chaotic steepest descent method (CSD method) [6] is used as chaotic retrieval method. This method chaotically itinerates among the local minimums in the energy function of the neural network. The retrieved patterns were restricted by the degree of chaos when the memorized patterns were chaotically searched by means of the CSD method in the Hopfield neural networks[12]. CFAMOUS has two functions. First, memorized patterns near the external input pattern are dynamically retrieved and this range is restricted by one parameter which defines the degree of system nonlinearity. Second, non-memorized and valid patterns can be retrieved as well as memorized ones. We explain the proposed evolutionary computation at the 6th chapter in detail.

3. The A-NET parallel computer

Baba et al. have been proceeding with development and research of an A-NET (Actors NETwork) parallel computer [8]. A-NET has a parallel object-oriented total architecture, and allows users to describe parallel programs naturally by using A-NETL (A-NET Language)[9]. The node processor on this computer consists of a processing element(PE), a local memory, and a router, and optional network topologies have been provided. A-NETL(A-NET Language) is a parallel object-oriented language which describes parallel programs naturally. A unit of parallel processing on A-NETL is an object. An object consists of data and procedure(method). On A-NETL, each object cooperatively sends or receives messages and processes them in parallel.

4. Multi-agent robots which move abreast

We will explain multi-agent robots which move abreast as a fundamental example of cooperative work in welfare robots. When two people walk together, they generally keep in step with one another. When turning, the outside person walks faster, while the inside person slows down. We have applied this conception to two robots which move abreast as shown in **Fig. 5**. In this model, a robot understands its situation, i.e., where the robot is, and it then carries out chaotic retrieval to adapt itself to its situation.

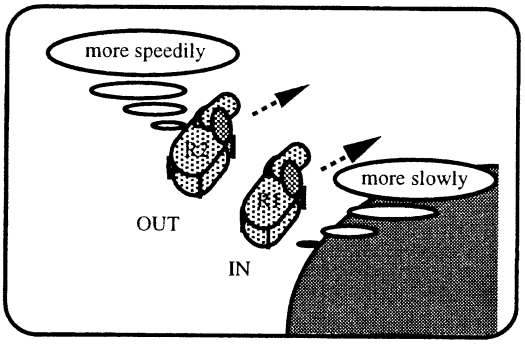


Fig. 5 Multi-agent robots which move abreast

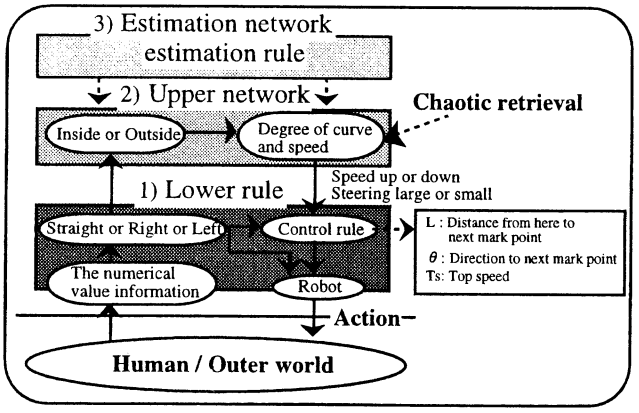


Fig. 6 Robot control block

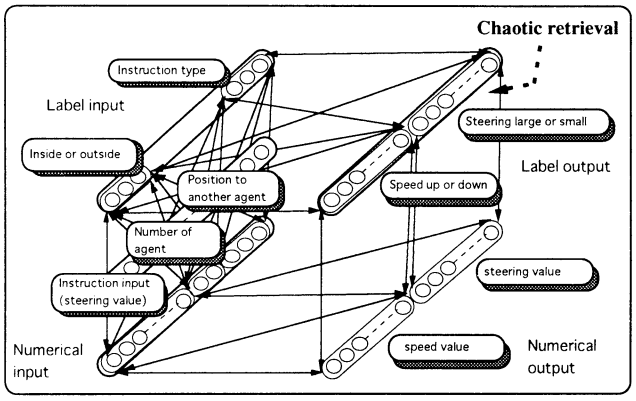


Fig. 7 Lower rule and upper network using CFS (Conceptual Fuzzy Set)

The robot control block is shown in **Fig. 6**. This control block is constructed based on the proposed agent model shown in **Fig. 2**. The upper network of this figure has the knowledge to adapt itself to the change in its situation. The output of the lower rule depends on this knowledge. The robots estimate their movement in the estimation network. These networks are realized by associative memory network based on FAMOUS or CFAMOUS. More concretely, the lower rule and the upper network are constructed by CFS (Conceptual Fuzzy Set)[11] as shown in **Fig. 7**. This CFS is constructed by hierarchical network applying (C)FAMOUS. The bottom of this figure shows the lower rule. The top shows the upper network. In the lower rule, the left is the numerical value input and the right is the numerical value output. The outputs are speed and steering value. In this application, a box of soft DNA is made by the CFS shown in **Fig. 7**. New soft DNA is created by chaotic retrieval on this CFS network. Each intelligence as box of soft DNA can evolve separately according to environmental information such as whether the mobile agent is inside or outside at turning abreast. In order to acquire suitable knowledge(i.e., good soft DNA) in the upper network, we use the proposed evolutionary computation. We create new generation of various agent robots with new knowledge by applying chaotic retrieval in the upper network.

5. ITS(Intelligent Transport System)

The ITS is the system which realizes safer and more efficient traffic and transportation by constructing the intelligent automobiles and road environment. For example, the realization of platooning of automobiles has been researched for that purpose. The platooning means platoon (i.e., group) running of automobiles. In platooning, if all platoons or all automobiles run based on the exactly same control knowledge, the traffic system may be a failure as a whole. Therefore, it seems that each platoon or each automobile needs to have the fluctuations of its intelligence in order to do the different movement from the others.

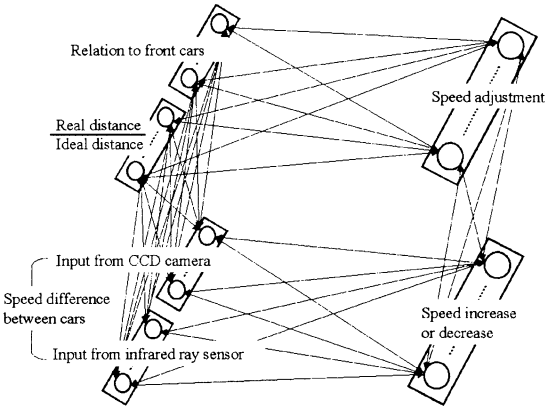


Fig. 8 Basic control block using CFS for ITS platooning

If we consider a platoon to be a group of automobile agents, we can apply the proposed evolutionary computation to this platooning. The basic control block for platooning is shown in **Fig. 8**. This control block is realized by CFS as well as the case of multi-agent robots which move abreast.

In ITS platooning, it seems that desirable intelligence of a automobile should be change according to whether the automobile runs as a head vehicle or as a middle vehicle or as a tail vehicle in the platoon. The situation of each automobile changes dynamically as the current middle vehicle changes to the head vehicle because of division of the platoon. Therefore, we apply evolutionary computation on soft DNA to each ITS automobile and aim to realize the dynamic development of intelligence.

6. Simulation results and discussion

6.1 Multi-agent robot which move abreast

First, we will explain the results of evolutionary computation based on chaotic retrieval and soft DNA in multi-agent robots which move abreast. Second, we will explain the results of parallel processing on agent robots. **Fig. 9** shows an example of the evolutionary process. The situation of this figure shows the case where the agents are given the "Turn Left" instruction. First, two agent robots can't move abreast because they move based on the same control value. The evolutionary computation creates new generation of various agents with new knowledge(i.e., the box containing new soft DNA) by applying chaotic retrieval in the agent's brain (i.e., associative memories). The generation consists of the group of the pairs of two agents which intend to move abreast. These pairs are kept or killed based on the estimation about their movement. Finally, the agent with the suitable knowledge(i.e., good soft DNA) to

move abreast is kept. Evolutionary computation is performed in the same way for all situations (i.e., for all instructions).

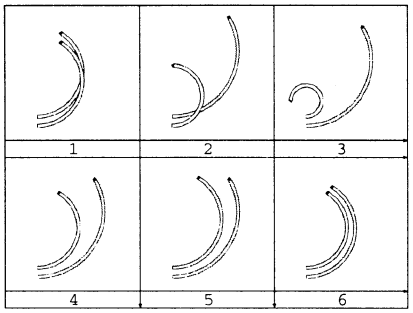


Fig. 9 An example of the evolutionary process

Table 1. The number of pairs of created agents in each instruction before the agent with suitable knowledge is acquired.
(The case of parallel algorithm of evolutionary computation)

Instruction	Right & Large	Turn Right	Right & Small	Left & Large	Turn Left	Left & Small
The number of pairs of created agents	26	5	10	27	6	9

Table 2. The number of pairs of created agents in each instruction before the agent with suitable knowledge is acquired.
(The case of serial algorithm of evolutionary computation : Total number is 67)

Instruction	Right & Large	Turn Right	Right & Small	Left & Large	Turn Left	Left & Small
The number of pairs of created agents	26	9	2	22	5	3

Table 1 shows a result in the case of parallel algorithm of evolutionary computation, that is, it shows the number of pairs of created agents in each instruction before the agent with suitable knowledge is acquired. Each knowledge item is acquired by creating the about 14 pairs of agents on average. In this parallel evolutionary process, each knowledge is created separately, that is, plural pairs of agents are created and perform evolutionary computation concurrently. Finally, all knowledges are integrated on the associative memory network of CFAMOUS. This integration of knowledge was realized by utilizing evolutionary computation using CFAMOUS. It is difficult for conventional neural networks to realize this evolutionary parallel computation.

After agent robots had acquired suitable knowledge for all instructions, we verified their movement by means of computer simulation. The results are shown in **Fig. 10**. As shown, the robots move abreast suitably when a series of instructions is given to the robots.



Fig. 10 The movement of robots (Instructions: Turn Left —> Right & Small —> Right & Large)

Table 2 shows a result in the case of serial algorithm of evolutionary computation, that is, it shows the number of pairs of created agents in each instruction before the agent with suitable knowledge is acquired. In this serial evolutionary process, one pair of agents repeats evolutionary computation serially to acquire all suitable knowledges.

In parallel algorithm, the maximum number of created pairs is 27 at the "Left & Large" instruction as shown in **Table 1**. The processing time of parallel algorithm is directly proportional to this number (27). In serial algorithm, the total number of created pairs through all situations is 67 from **Table 2**. The processing time of serial algorithm is directly proportional to this total number (67). Therefore, the parallel algorithm of evolutionary computation is about 2.5 times faster than its serial algorithm from the viewpoint of the number of pairs of created agents.

We also realized parallel program which simulates the associative inference and the movement of robots in multi-agent robots which move abreast in parallel, and implemented it on A-NET parallel computer. This program simulates the movement of two agent robots which move abreast concurrently and also simulates parallel associative inference in the brain of each agent. The simulation results show that the parallel program is about 11 times faster than conventional serial program. The above results indicate the usefulness of combining parallel processing and proposed evolutionary computation based on chaotic retrieval.

6.2 ITS(Intelligent Transport System)

We apply the proposed evolutionary computation on soft DNA to the platooning of ITS (Intelligent Transport System). In this application, each box of soft DNA is made by the CFS shown in **Fig. 8**, and each box develops the intelligence for each role such as the roles of a head vehicle, a middle vehicle and a tail vehicle in the platoon according to environmental information. We show the soft DNA is useful for the realization of dynamic cooperation by simulations.

Furthermore, we show the proposed evolutionary computation improves the control performance about distance between vehicles in the ITS platooning.

Fig. 11 shows the simulation image of ITS platooning. Simulation result showed that the average error between the ideal distance and the actual one improved about 1/4 by the evolved soft DNA.

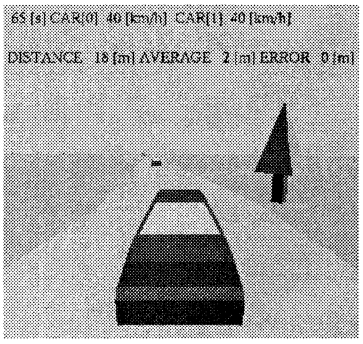


Fig. 11 ITS simulation image

In biological DNA, an individual of actual life can be developed from a set of the minimum essential DNA (called "Genom") theoretically. Similarly, in soft DNA, we aim to be able to develop various patterns of intelligence dynamically according to the environmental information from the minimum essential description of intelligence. After this, we will introduce some biological ideas into our chaotic evolutionary computation. At first, we will prepare some groups of agents and give each group the different conditions from other groups and carry out the evolution of each group, like the actual life evolves variously in various environments. Second, we will give the agents their lifetimes like the actual life has it. Moreover, we aim to introduce some kind of recursive process into soft DNA to be able to apply the developmental process repeatedly. Hence, soft DNA will be able to cope with the fractal structure of the systems in the real world.

7. Conclusion

This paper proposed an evolutionary parallel collaborative computation by chaotic dynamics and Soft DNA instead of conventional GA (Genetic Algorithm) for such intelligent agents as welfare robots which assist humans. This evolutionary computation was realized by applying chaotic retrieval on associative memories. We applied this evolutionary computation to multi-agent robots which move abreast. Essentially, the process of this evolutionary computation is parallel processing. Therefore, we implemented its parallel processing algorithm on A-NET (Actors Network) parallel object-oriented computer, and showed the usefulness of parallel processing for proposed evolutionary computation.

References

[1] T. Hoshino: Dream and Distress of Alife, SHOUKABOU (1994) (in Japanese).

[2] H.Nakahara and T.Sagawa: Virus theory of evolution, HAYAKAWASYOBOU (1996) (in Japanese)

[3] W.-Z.Xu, A.Kashiwagi, T.Yomo and I.Urabe: Fate of a mutant emerging at the initial stage of evolution, Researches in Population Ecology, 38(2), pp231-237 (1996)

[4] N.Kohata, T.Yamaguchi, Y.Wakamatsu and T.Baba: Evolutionary Parallel Computation based on Chaotic Retrieval and Creation, Proceedings of the 4th International Conference on Soft Computing (IIZUKA'96), Vol.2, pp.638-641 (1996)

[5] T.Yamaguchi: Fuzzy Associative Memory System, Journal of Japan Society for Fuzzy Theory and Systems, Vol.5, No.2, 245-260 (1993) (in Japanese).

[6] J.Tani: Proposal of Chaotic Steepest Descent Method for Neural Networks and Analysis of Their Dynamics, Trans.IEICE, Vol. J74-A, No.8, pp1208-1215(1991).

[7] B.Kosko: Adaptive Bidirectional Associative Memories, Applied Optic, Vol.26, No.23, pp.4947-4960(1987).

[8] T.Baba, T.Yoshinaga, Y.Iwamoto and D.Abe: The A-NET *Working* Prototype: A Parallel Object-Oriented Multicomputer with Reconfigurable Network, Proc. International Workshop on Innovative Architecture for Future Generation High-Performance Processors and Systems (IWIA'97), IEEE Computer Society Press, pp.40-49 (1998).

[9] T.Baba and T.Yoshinaga: A-NETL: A Language for Massively Parallel Object-Oriented Computing, Proc. of Programming Models for Massively Parallel Computers, IEEE Computer Society Press, pp.98-105 (1995).

[10] T.Sato, H.Ushida, T.Yamaguchi, A.Imura and T.Takagi: Chaotic Memory Search in Fuzzy Associative Inference, Proc. of IIZUKA'94, pp.203-206(1994).

[11] T.Takagi, T.Yamaguchi and M.Sugeno: Conceptual Fuzzy Sets, International Fuzzy Engineering Symposium'91 (IFES'91), Vol.2, pp.261-272 (1991).

[12] J.J.Hopfield and D.W.Tank: Neural computation of decisions in optimization problems, Biological Cybernetics, 52, pp.141-152, 1985.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Development of an Intelligent Data Carrier (IDC) System and its Applications

Daisuke Kurabayashi† Hajime Asama† Teruo Fujii‡ Hayato Kaetsu‡ Isao Endo‡
† Department of Research Fundamentals Technology
‡ Biochemical Systems Laboratory
The Institute of Physical and Chemical Research (RIKEN)
Hirosawa 2-1, Wako-shi, Saitama, 351-0198 JAPAN
{*daisuke asama fujii kaetsu endo*}*@cel.riken.go.jp*

Abstract

This paper introduces a system, including a new device named "Intelligent Data Carrier (IDC)" for emergent cooperative behaviors by autonomous robots. The concept, system design and applications are presented. The IDC is an immobile but portable device for information storage and management. The IDC system is composed of a reader/writer which will be attached onto a robot and a tag which will be carried and located by a robot. A tag has a CPU, memory and a battery for communication and data management. The IDC realizes local communication by FM radio wave. The IDC progresses the flexibility of the multiple robot system by reducing the amount of global communication for mutual exchange of information. In the paper, we introduce some application examples to show effectiveness of the IDC system in the practical missions of multiple autonomous robot systems.

Key words: Intelligent Data Carrier, Cooperation of Autonomous Robots

1 Introduction

In order to realize cooperative behaviors among multiple robots, it is important for the robots to be equipped some communication devices to exchange data between robots and to obtain information about tasks and working areas. Some studies suggest that the local communication systems are effective for multiple autonomous robot systems (Hara[1]-Yoshida[6]). In former studies, robots only try to adjust themselves to the environmental changes or unexpected situation. However, social insects such as ants arrange the environments with laying a chemical trail (pheromone). They indirectly exchange acquired information. By the analogy of the creatures, we have found that a device which enables the robots to leave and exchange the information in their working areas should be effective for their cooperative behaviors.

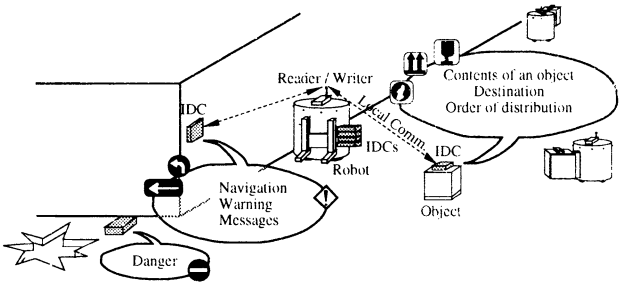


Fig. 1: The overview of the IDC system.

In this paper, we introduce a system including a new device named the "Intelligent Data Carrier: IDC" which is an immobile but portable medium for information storage and management in collective robot systems. By using the IDC system, the robots in the system can efficiently acquire and store the knowledge of the working environment without consuming global communication resources. The detailed description of the IDC system is presented in the following chapters. In this paper, the effectiveness and the applicability of the system are demonstrated by applying the IDC system to the selected problems in a certain kind of tasks by autonomous robots.

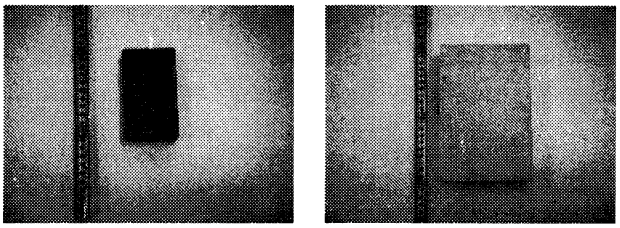
2 Development of the IDC system

2.1 Overview of the IDC System

The authors have developed the IDC to reduce the traffic of global communication by providing local communication links and local information management functions. By reading information from and writing it into the IDCs, the robots can use the IDCs as media for inter-robot indirect communication. Furthermore, by putting the IDCs on some specific locations in the working environment, the robot can allocates functions to act as agents for information storage and management(Fig. 1).

The IDC system consists of portable information

storages (tags) and reader/writer devices carried by the robots(Fig. 2). We usually call the tags "IDCs". The reader/writer which is mounted on a robot plays an active role in the system by initiating communication. The reader/writer has a serial interface connecting to a computer on a robot. The tags are usually placed in an environment or on objects to act as local information storage. They store and process information and reply to requests from a reader/writer, but don't initiate communication by themselves. The tag usually sleeps to save their energy until it receives communication or processing request from the reader/writer.



(a) A tag (b) A reader/writer

Fig. 2: The prototype of the IDC system

2.2 Specifications

The specifications of the IDC are shown in table 1. The IDC is an application of RF-ID (Radio Frequency IDentification) type low intensity electromagnetic wave transmission. Their transmission frequency is 300[MHz] and the communication range is up to 2.0[m]. The speed of data transfer is 1200[bps].

Table 1: Specifications of the IDC

Media	Electromagnetic wave
Frequency	300 [MHz]
Modulation	FM
Data rate	1200 [bps]
Power Source	a Li-ON battery (3V)
Size (tag)	110 x 65 x 25 [mm]
(reader/writer)	195x130x50 [mm]

We have prepared universal data sequences to call the IDC, to write down data onto the IDC, to read the data from the IDC, and to conclude the communication. A frame check sequence is also attached to each data packet to keep the consistency of the data reading frame. The procedure of the communication is as follows. When an IDC exists near the robot, the IDC responds to the robot's call and the communication will be established. However, when the IDC does not respond, time-out sequence will be executed and the next calling sequence for the IDC will start up.

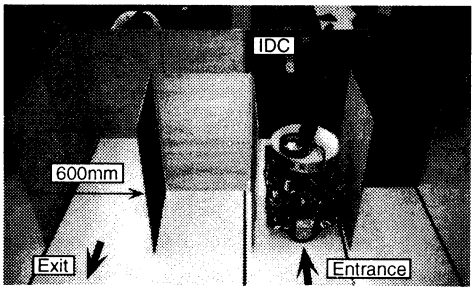


Fig. 3: Experimental setup of a corridor.

In the following chapters, we will describe two types of applications in which the IDC system is utilized by autonomous mobile robots. One is autonomous information acquisition and sharing. In the application, we will show that the IDC system accelerates the efficiency of cooperation among autonomous robots. The other is distributed management of a map. We will denote that the IDC system improves the robustness of autonomous mobile robots.

3 Autonomous information acquisition and sharing

Abilities of autonomous information acquisition and sharing are essential for autonomous robot systems. When a robot obtains some information about a part of its work area, it can leave the information with the IDC. In other words, the work area acquires local information in itself. Other robots can share the information without global communication with the IDC. The IDC system realizes spatial distribution of the robots, which leads to comprehensive recognition of the work area in a quite effective way by multiple autonomous robots. To demonstrate such kind of advantages, we attempted to implement the IDC system for cooperative exploration tasks in indoor corridor environment by multiple autonomous mobile robots.

3.1 Experiment of cooperative corridor exploring

3.1.1 Experimental setup

Figure 3 shows the indoor corridor setup for the experiments. The corridor has a branch which leads to a dead-end. We applied autonomous omni-directional mobile robot (Asama[7]) which is capable of dead reckoning navigation based on odometry and a gyro. By eight infra-red sensors the robot is able to detect the configuration of the corridor system. The robot knows its absolute direction by its gyro. At the initial state, the IDC has no information.

For this exploration task, we set a simple algorithm

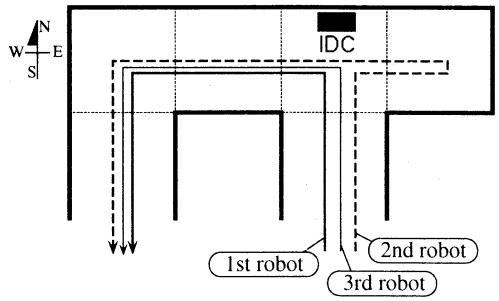


Fig. 4: Trajectories of the robots.

Table 2: Transition of data in the IDC.

(a) After the first trial.

Direction	Knowledge	Status
N	Wall	Finished
W	Path	Searching
S	Path	Finished
E	Path	NotYet

(b) After the second trial.

Direction	Knowledge	Status
N	Wall	Finished
W	Path	Searching
S	Path	Finished
E	DeadEnd	Finished

into each robot. The algorithm is composed of following rules.

(1) Go forward until finding a junction.
(2) Ask the way to an IDC (tag). If the IDC suggests a direction, follow it. If the IDC has no enough data, explore unknown direction.
(3) Tell selected direction to an IDC when it leaves the junction.

3.1.2 Experimental Result

Figure 4 shows the trajectories of three robots during the exploration task and table 2 shows the stored data in the IDC placed at the junction in the corridor. The first robot chose left way randomly because the robot obtained no information from the IDC. Then the IDC knew that a robot went to left. When the second robot asked to the IDC about the way, it knew that no robot had explored the right way. On the basis of the rule, the robot entered the branch and found the dead-end by its infra-red sensors. The robot returned to the junction and told to the IDC about the dead-end. After exploration by the two robots, the third robot acquired information about the dead-end of the branch by the IDC. Thus the robot can decide its direction.

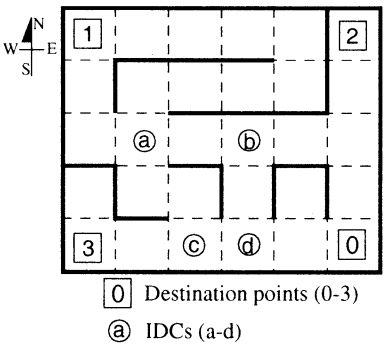


Fig. 5: A work area for the robots.

3.2 Quantitative evaluation by simulations

3.2.1 Settlement of simulations

In this section, we evaluate the quantitative efficiency of the IDC system by simulations. We set an iterative transportation task. Figure 5 shows the working area of the iterative transportation. A mobile robot moves one grid per step. There are four destination points (numbers with squares). Robots engaged in the task move to ordered destinations. A robot is ordered only a number of the destination points. We assume that a robot can distinguish the numbers of the destination points and it can detect its absolute orientation by its gyro, but it can not know its position. Thus it can not refer to a map. It has to decide its direction by itself when it passes a junction. At the initial state, a robot selects the direction randomly. We added four IDCs at the positions in the working area and they are indicated by alphabets with circles. The algorithm to utilize the IDCs will be described later.

3.2.2 Algorithm to utilize the IDCs

The algorithm is composed of two parts: algorithm of data storage and that of decision making at the junctions. At the initial state, no IDC has knowledge of the environment. Although a robot has no map, it can remember the number of the latest destination which it has visited. A robot tells an IDC the direction which it comes from and the number of the latest target when it entered a junction. The IDC counts the told numbers of destinations at each direction. After a moment, the IDC stores data about the directions of destinations based on the amount of obtained information. Then a robot can get the most hopeful destination by communicating with an IDC of a junction. The flow of the algorithm is as follows.

(1) If a robot can not find any IDC (or the IDC has no data), it decides its direction at random.

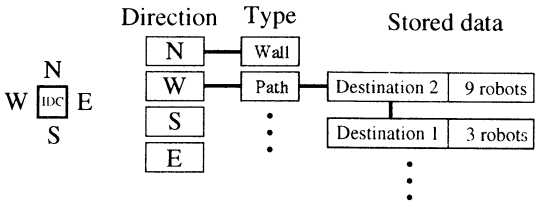


Fig. 6: Data structure in an IDC

(2) When a robot can find an IDC which has valid data, it decides its direction according to the ratio of numbers of destination points stored in the IDC.

(3) Independent of existence of IDCs, a robot decides its direction randomly at a constant ratio.

We set the constant ratio as 10%.

3.2.3 Simulation results.

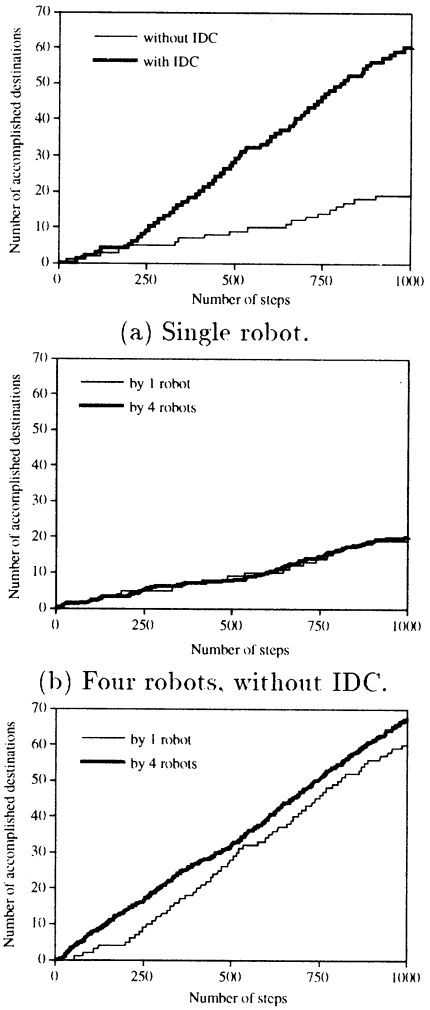
We count the number of achievement of destination points as efficiency evaluation during 1000 steps working of robots. First, we compared cases of with and without IDCs when single robot was in charge of the task. Figure 7(a) shows the numbers of achieved destinations. By the figure, we can find that the IDC system advanced the efficiency of the iterative transportation about 150% more than that of the case without IDCs. Next, we applied four robots to the same simulations. Figure 7(b) shows the numbers of achieved destinations. The line of four robots denotes the averaged value of the four robots to compare with that of single robot. Without IDC, there are no progress of the efficiency. However, robots improved the efficiency more than 10% by utilizing the IDC system (Fig. 7(c)), because the acquisition of information was accelerated.

4 Distributed management of a map

Recently, many researchers and developers have been trying to apply autonomous robots to tasks in actual environments. In former studies, users have to make exact model or map of a task and robots. However, it is difficult to make precise model when we apply robots to natural environments.

Here, let us suppose a task of removing weeds in a forest. A robot visits trees and cuts weeds around them. It is quite difficult to make a map of trees exactly. From the view point of mobile robots, it is no use moving according to global map because the accuracy of motions in natural environment can not be so high.

In this chapter, we propose distributed management of a map. We distribute the contents of a map into



(a) Single robot.

(b) Four robots, without IDC.

(c) Four robots, with IDCs.

Fig. 7: Comparison of task execution

many local maps managed by IDCs. By the IDC system, a robot can obtain local map information where it wants. By the identification of trees and adjustment of its position to the local map information, a robot can avoid storing its motion errors (Fig. 8).

4.1 Expression of the local map by Delaunay diagram

At first, we set following assumptions.

- An environment in which a robot works has a boundary.
- A robot has to visit all trees in the environment.
- We attach an IDC which contains local map information to each tree in advance.

Although we can hardly obtain global map information in a natural environment such as a forest, we can easily measure relative position between neighbor-

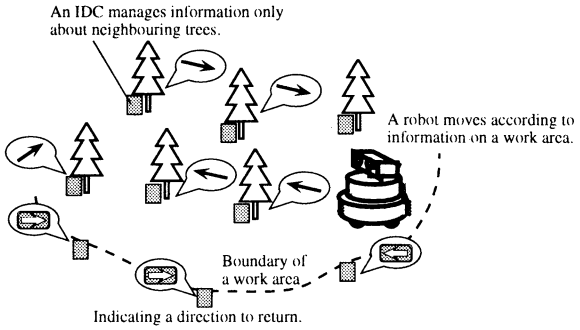
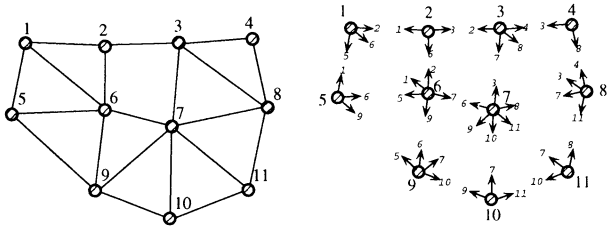


Fig. 8: Conceptual view of local map management.

ing trees. Now we have to discuss how to determine the locality of a map. In the field of computational geometry, we can find the Voronoi diagram (Okabe[8]) which gives division of a plane according to distances from some points (called "generators"). The Delaunay diagram shows a network of neighboring cells in the Voronoi diagram. The Delaunay diagram is feasible to the local map expression though the Voronoi diagram has same amount of mathematical information as the Delaunay diagram, because it is a simple diagram. Thus we generate local maps based on the algorithm of the Delaunay diagram. Figure 9(a) shows an example of the Delaunay diagram. Circles with numbers indicate targets in the environment. Targets which are connected by an edge are "neighbors". Thus we describe relative positions of neighbors onto each local map of a target (Fig. 9(b)).



(a) Delaunay diagram (b) Local expression

Fig. 9: Expression of local map

We use polar coordinates to denote relative position in a local map, because a mobile robot has two different types of motion error. One is an error of distance, and the other is an error of orientation. By the polar coordinate expression, a robot can refer to the local map independently about its running distance and current orientation.

Figure 10 shows the architecture of a local map. The order to visit trees is set in advance. If there is no difference between the maps and actual environment, a robot needs only relative positions of next tree. However, there can be many differences, a local map of a tree contains data of all edges of the Delaunay diagram. The local maps provide information for recovering when a robot fails to get contact with current target.

Table 3: Assumptions for simulations

A mobile robot	
Size	A circle: R = 0.3 [m]
Sensing range	0.7[m]
Communication range	1.0[m]
Motion errors	±5%
The environment	
Layout of trees	About 1.8[m] grid
Settlement of IDCs	On each tree
The order to visit trees	Given in advance

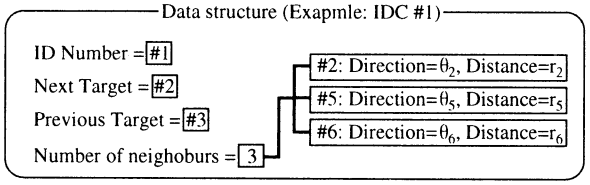


Fig. 10: Data structure in an IDC

4.2 Robustness of the distributed management of a map

4.2.1 Settlement of simulations

Here we made simulations to verify effectiveness of the local map management. At first, we set assumptions as table 3 according to actual robots and forests.

We made simulations in two types of errors: quantitative error and topological error of the local maps. We will show that an autonomous robot can override those errors.

4.2.2 With a quantitative error

In the first case, a map which was set to the IDCs had quantitative error. In Fig. 11, circles denote trees (targets) with IDCs. Figure 11(a) shows ideal map on which we generated local maps based, and (b) indicates actual layout of the targets. The tree with hatching in (b) was located at different position from the ideal map. The broken lines show a trajectory of a mobile robot. In Fig. 11(b), the robot once failed to reach next tree of the hatched tree, because the local map of the hatched tree suggested wrong position according to its ideal position. However, referring to other local maps of trees around the hatched tree, the robot found the way (one of edges of the Delaunay diagram), then it reached the tree and continued its task.

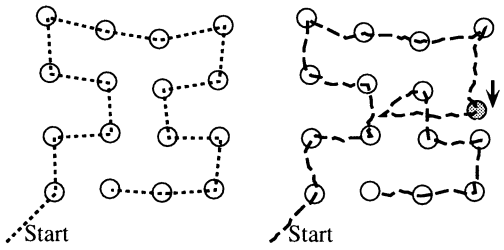


Fig. 11: A case with a quantitative error.

4.2.3 With a topological error

In the next case, a tree is disappeared. It causes topological difference from the ideal map. When a robot reached the position where the local maps of trees indicate the disappeared tree, it looked for the tree. After several trials, the robot decided that the tree was disappeared. Then the robot restores the local maps of trees around the disappeared tree. Because the local maps are generated based on the algorithm of the Delaunay diagram, the robot can restore the local maps according to information contained by the local maps around the disappeared tree. The triangles which are formed by the Delaunay diagram have many geometrical characteristics. Delaunay triangles minimize their circumscribed circles. Utilizing the characteristic, a robot can restore the local maps around the disappeared tree in the following steps.

(1) Determine a polygon which surrounds the disappeared tree. A robot should only restore Delaunay diagram in the polygon because there is no changing outside it.
(2) Select an edge of the polygon which is not currently an edge of Delaunay diagram.
(3) Search a vertex of the polygon which forms a triangle with the edge whose circumscribed circle contains no vertex inside.
(4) Generate new Delaunay edges by the triangle.
(5) Go to (2) until all vertexes on the polygon are connected to Delaunay edges.

We can judge that a vertex (x_4, y_4) is outside of a circumscribed circle of three vertexes (x_1, y_1),(x_2, y_2),(x_3, y_3) by equation (1).

$$\begin{vmatrix} x_1 & y_1 & x_1^2+y_1^2 & 1 \\ x_2 & y_2 & x_2^2+y_2^2 & 1 \\ x_3 & y_3 & x_3^2+y_3^2 & 1 \\ x_4 & y_4 & x_4^2+y_4^2 & 1 \end{vmatrix} < 0 \qquad (1)$$

The order of the calculation cost is $O(n \log n)$. It is small enough for autonomous mobile robots to calculate.

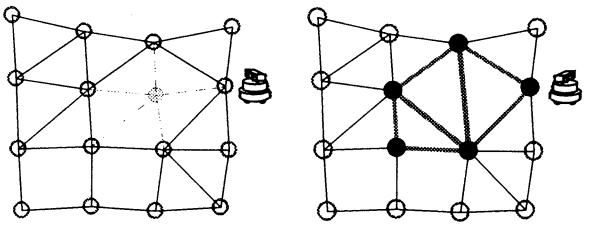


Fig. 12: A case with a topological error.

5 Conclusion

In this paper, we have described a newly developed device called the Intelligent Data Carrier (IDC). A multiple robot system can reduce global communication by utilizing the local communication realized by the IDC system. We have shown that the IDC system improves effectiveness of cooperation and robustness of autonomous robots by experiments and simulations.

References

1. Hara F, et al.(1992), Effects of Population Size in Multi-Robots Cooperative Behaviors, Proc. Int. Symp. on on Distributed Autonomous Robotic Systems (DARS'92), pp. 3-9.
2. Ueyama T, et al.(1993), Self-Organization of Cellular Robots Using Random Walk with Simple Rules, Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'93), pp. 595-600.
3. Wang J, et al.(1994), On Sign-board based Inter-Robot Communication in Distributed Robotic Systems, Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'94), pp. 1045-1050.
4. Arai T, et al.(1993), Information Diffusion by Local Communication of Multiple Mobile Robots, Proc. IEEE Int. Conf. on Systems, Man and Cybernetics (SMC'93), Vol. 4, pp. 535-540.
5. Yoshida E, et al.(1994), Effect of Grouping Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS'94), pp. 808-815.
6. Yoshida E, et al.(1995), A Design Method of Local Communication Range in Multiple Mobile Robot System, Proc. IEEE Int. Conf. on Intelligent Robots and Systems (IROS'95), pp. 274-279.
7. Asama H, et al.(1995), Development of an Omni-Directional Mobile Robot with 3 DOF Decoupling Drive Mechanism, Proc. IEEE Int. Conf. on Robotics and Automation (ICRA'95), pp. 1925-1930.
8. Okabe A, et al.(1992), Spatial Tessellations - Concepts and Applications of Voronoi Diagrams, John Wiley & Sons, Chechester.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Physical Agent and Media on Computer Network

Yasuharu KUNII*, Hideki HASHIMOTO**

* Faculty of Science and Engineering, Chuo University
1-13-27 Kasuga, Bunkyo-ku, Tokyo 112-8551, Japan.
** Institute of Industrial Science, University of Tokyo
7-22-1 Roppongi, Minato-ku, Tokyo 106, Japan.
E-mail: kunii@elect.chuo-u.ac.jp, hashimoto@iis.u-tokyo.ac.jp

Abstract - In this paper, we introduce our two experimental systems and experimental results of Robotic Network System(RNS). One is a mobile-robot: MUSEAR (MUlti-SEnsored Autonomous Rover) which behaves as a physical agent of an operator in the remote site. The operator can drive MUSEAR as him or her agent, watching a real-time image from it's on-site camera in the remote site. The other is HandShake Device (HSD) which is a haptic interface using tele-operation and force-feedback techniques to display physical information in the network. This system is to be a test-bed of Tele-operation and defines as a physical media using a force information.

1 Introduction

Recently Multi-media and Internet have the eyes of a world. Many kinds of information run through the network. However information on the network can not have a physical influence to our real environment. On the other hand, robots can expand an ability of multi-media to make a physical influence. They can behave as a physical media or agents which have an effect on the physical environment through the network.

With the spreading availability of high bandwidth networks, B-ISDN connections will be broadly available early in the next century. We currently build a system called Robotic Network System (RNS) in which robots can be physical agents as one of media. In such a system various types of robots are connected to this network in different places such as homes, schools, hospitals, constructions, fire and police stations. They can be branched into our daily life.

In this paper, we introduce our two experimental systems and experimental results of RNS. One is a mobile-robot: MUSEAR (MUlti-SEnsored Autonomous Rover) which behaves as a physical agent of an operator in the remote site. The operator can drive MUSEAR as him or her agent, watching a real-time image from it's on-site camera in the remote site. The other is HandShake Device (HSD) which is a haptic interface using tele-operation and force-feedback techniques to display physical information in the network. This system is to be a test-bed of Tele-operation and defines as a physical media using a force information.

2 Robot as Physical Agent

We can access a remote environment through the computer network by using a robot. It will be expand the limitation of human ability for example: the time and physical constraints. In this section, for the first step of our research, tele-driving experimental results of our mobile robot: MUSEAR is shown.

2.1 MUSEAR: Physical Agent

Our mobile robot, MUSEAR(MUlti SEnsored Autonomous Rover), is shown in Fig.1. MUSEAR is defined as a home robot in RNS and it's based on an electric wheel chair. It can be autonomous navigation with Note type PC and some sensors, for example, ultra-sonic sensors. This system communicates to the other computer system through the Internet by a wire-less LAN and get cooperations by them, for example, map information.

2.2 Tele-Driving System and Experimental Results

In the experiment of Tele-Driving, MUSEAR is connected to the host through the inter-net between Institute of Industrial Science (Roppongi in Tokyo) and Waseda University. We have two agents in that time. One is our system: MUSEAR. The other is Mini Planetary Rover (MPR) of Meiji University. Each system can recognize each other through a camera on each system. The experimental system structure of MUSEAR is shown in Fig.2. An operator uses an analog joystick to control a velocity vector watching a color video image transmitted by tele-conference system. We use Resource Reservation Protocol(RSVP) to keep data speed (64Kbps) on some network routers. Figure3 (a) and (b) show a photograph of our experiment and it's control screen. We have an interaction with MPR in the remote environment. Operators of each system can recognize each other in Fig.3(b) from MUSEAR and (c) from MPR.

In this experiment, we smoothly controlled systems in the morning but not the afternoon because of communication load. Difficulty of operation is mainly caused by a communication delay of image data. Time delay of image data is more important than of command data because time delay is not so large in this case. So we need to improve GUI with some assistance on the host.

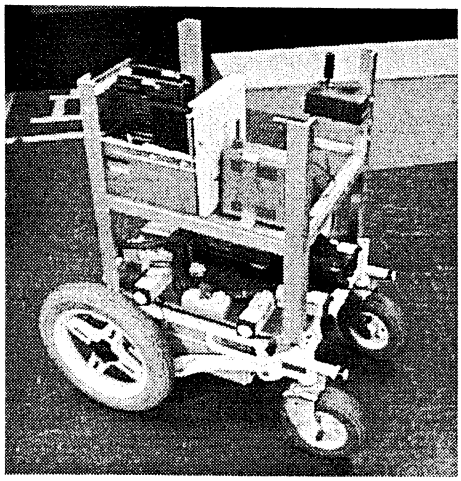


Figure 1: Multi-Sensored Autonomous Rover: MUSE-AR

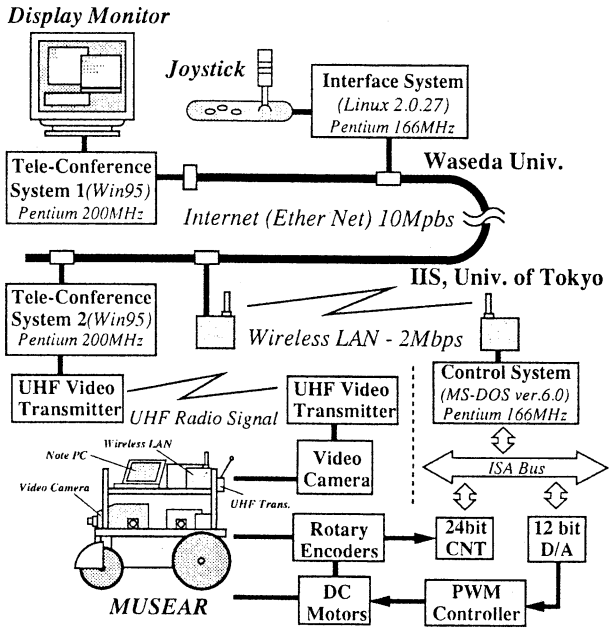


Figure 2: Experimental System for Tele-drive

Variable time-delay is also one of the reasons of difficulty. Variable time delay always exists on the network. It will be one of future works.

3 Robot as Physical Media

It can be possible to transmit physical information from remote site by displaying a force and a torque information using the network and robot technology. Specially a haptic interface, as a master system of tele-operation, is to be a media to display users an environment in a remote site and a computer system by force information.

3.1 HandShake Device : Physical Media

HSD is an interface device that transfers the motion of the shake-hand partner to the user. Therefore HSD represents the hand of the other user as shown in Fig.4 and 5.

Fig.6 shows the structure of HSD. HSD is composed of two arms: the master arm and the slave arm. The master arm is grasped by the user while the slave arm



(a) Operator



(b) Image from MUSEAR



(c) Image from MPR of Meiji Univ.

Figure 3: Experiment of Tele-Control

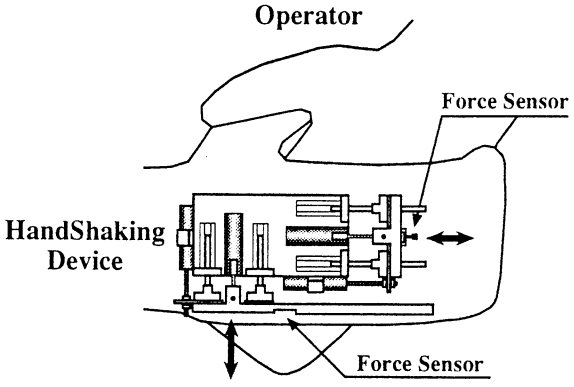


Figure 4: Handshake through HSD.

represents the handshaking action done by the other user (handshake partner). Each arm is connected to the shaft of the linear motion motor. The linear guiders support linear motion of the arms.

Positions of the master and slave arms are measured by linear-motion potentiometers. Force information is acquired via a strain gauge bridge connected to each arm(Fig.7).

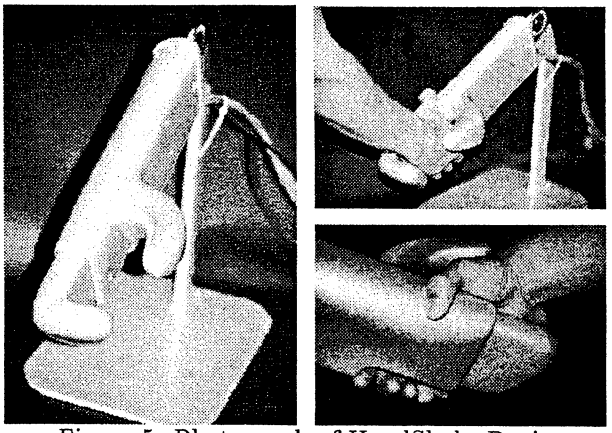


Figure 5: Photograph of HandShake Device

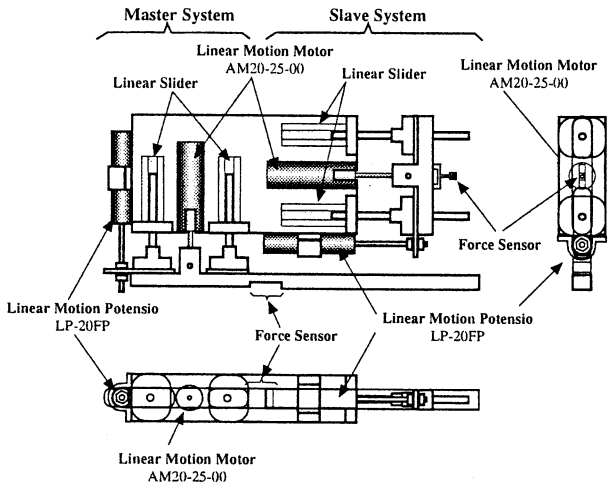
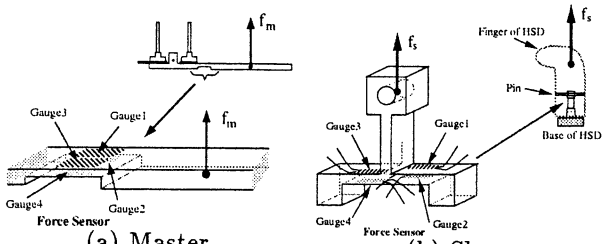


Figure 6: Schematic of HandShake Device



(a) Master (b) Slave

Figure 7: Schematic of Force Sensor

3.2 System Structure of HSD

Tele-handshaking system containing two HSDs (one is located at site A and operated by Operator A, the other is located at site B and operated by Operator B) is composed of two single degree-of-freedom (DOF) linear motion master-slave systems: 1) the master arm A and

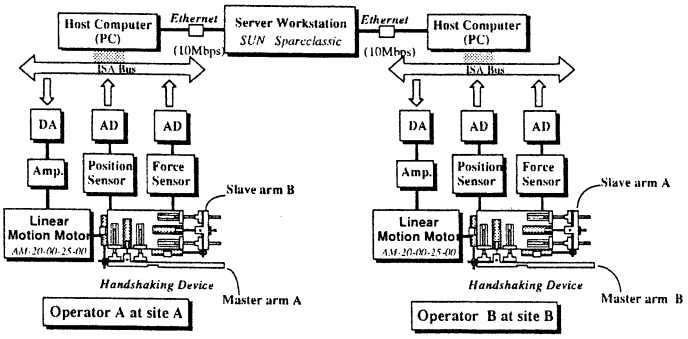


Figure 8: System Structure of HSD

the slave arm A which are operated by the operator A, 2) the master arm B and the slave arm B which are operated by the operator B. The structure of the tele-handshaking system is illustrated in Fig.8.

In each site, position and force information from position and force sensors are sent through an AD board to a host computer (PC Pentium 100 MHz). The controller is realized in the host computer. The controller generates control signals which are sent through a DA board and an amplifier to the motors in order to control the motion of the master and slave arms.

Data between two sites is communicated through the Internet. The data from one host computer is transmitted to the other host via a server workstation and vice versa. The server workstation can be located at some place in the network in order to set node for passing data when it is sent through the Internet by the most standard network protocols TCP/IP.

3.3 Control Algorithm for HSD

3.3.1 Virtual Impedance

Based on the concept of the impedance control, a technique of compliant motion for a teleoperator by using a virtual impedance (VI) (or virtual internal model or virtual model) has been used [7][8][6].

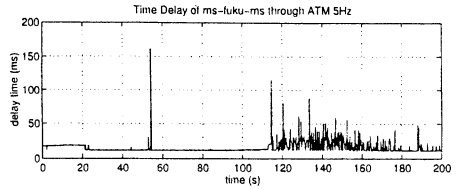
It is assumed that an end-point (or end-effector) of a manipulator arm in a virtual compliance frame is controlled in position depending on the virtual impedance of the frame. VI is composed of a virtual spring (or compliance), a virtual damper, and a virtual mass. These parameters may be programmed and/or be different in direction, or change with time in order to produce good maneuverability.

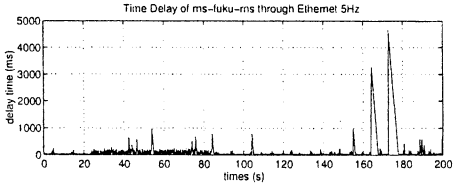
In the case of a 1 DOF linear motion manipulator, it is assumed that a virtual mass, M_v, is attached to a manipulator arm. This virtual mass is supported by a virtual damper, D_v, and a virtual spring, K_v.

When a force is applied to the virtual mass, the position of the virtual mass is modified according to the dynamics of VI. The dynamic equation of VI is as follow.

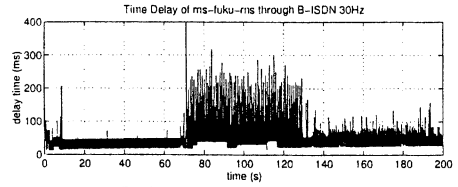
$$F(t) = M_v \ddot{x}(t) + D_v \dot{x}(t) + K_v x(t) \qquad (1)$$

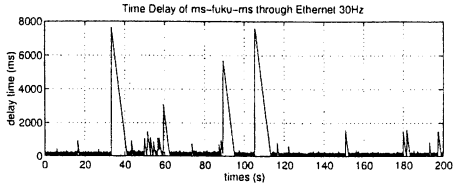
where $F(t)$ is the force detected at the virtual mass and $x(t)$ is the displacement of the virtual mass.



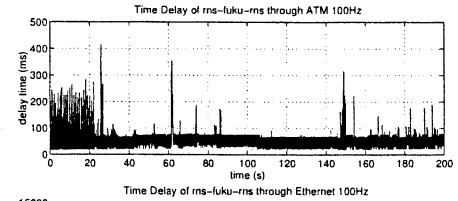


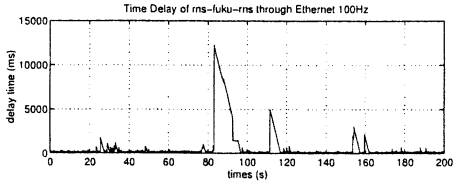
(a) 5 [Hz]





(b) 30 [Hz]





(c) 100 [Hz]

Figure 9: One-way time delay in data trasmisson from Tokyo to Nagoya and back to Tokyo

3.3.2 Compensation of Time-delay

Communication time-delay comes out on tele-operation system. Furthermore the inter-net has variable time-delay because of network load by it's own anonymous users.

Variable time-delay on the inter-net(Ether) and the B-ISDN (ATM network) between Tokyo and Nagoya are shown in Fig.9. Some large time delays can be recognize in Fig.9. The time-delay of ATM network is better than Ether-net, however, it still has 100[ms]. Variable time-delay might be an important problem for tele-operation on computer network. Here, variable time-delay will be a future work and we assumed that time delay is constant in this research.

A new control approach, *virtual impedance with position error correction (VIPEC)*, is proposed. This new control approach is an extension of the control approach by Otsuka et al. [8]. The control diagram of a teleoperator with time delay utilizing the VIPEC is illustrated in Fig.10. The PEC part (the shaded part) is added to the original approach in order to improve the response of the teleoperator by reducing the position error between the master and slave positions x_m, x_s.

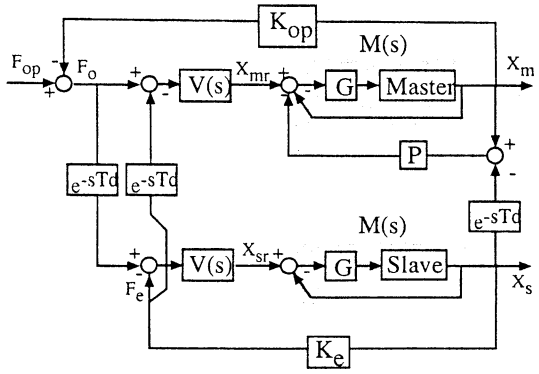


Figure 10: Control System Diagram with VIPEC

3.3.3 Modeling of HSD

Fig.10 shows the control system diagram of the master-slave system inside the tele-handshaking system with VIPEC approach. The armature inductance, L_a of each motor is neglected. The two linear motion motors represented by the master block and the slave block can be modeled as simple DC-motors with the transfer function

$$C(s) = \frac{X(s)}{E_a(s)} = \frac{K_m}{s(T_m s + 1)} K_p \qquad (2)$$

where X is the displacement of the motor shaft, E_a is the applied armature voltage, K_m is the motor gain constant, T_m is the motor time constant, and K_p is a constant to convert the angular displacement into displacement. K_m and T_m are obtained from

$$K_m = \frac{K}{R_a b + K K_b}, \quad T_m = \frac{R_a J}{R_a b + K K_b} \qquad (3)$$

where K is the motor-torque constant, R_a is the armature resistance, K_b is the back emf constant, J is the moment of inertia of the motor, and b is is the viscous-friction of the motor is very small and can be neglected.

The servo controller is a proportional controller whose gain is G. The closed-loop transfer function of the manipulator arm and the controller is

$$M(s) = \frac{C(s)G}{1 + C(s)G} = \frac{\psi}{\alpha s^2 + \beta s + \gamma} \qquad (4)$$

where $\alpha = 1$, $\beta = \frac{1}{T_m}$, and $\gamma = \psi = \frac{G K_m K_p}{T_m}$.

Gain P represents PEC gain. For simplicity, the environment is modeled by a spring with stiffness constant K_e. Therefore the dynamics of the spring interacting with the slave arm is modeled as follow.

$$F_e(t) = K_e x_s(t) \qquad (5)$$

The dynamics of the operator including the dynamics interaction between the operator and the master arm is approximated by

$$F_{op}(t) - F_o(t) = M_{op}\ddot{x}_m(t) + D_{op}\dot{x}_m(t) + K_{op}x_m(t) \quad (6)$$

armature resistance, R_a	10.9Ω
armature inductance, L_a	neglected
viscous friction, b	neglected
moment of inertia, J	$2.08 * 10^{-7} kg - m^2$
back emf constant, K_b	$0.034V/(rad/s)$
motor-torque constant, K	$0.06Nm/A$
Motor time constant, T_m	$0.0011s$
Controller gain, G	10000
Motor gain constant, K_m	$29.41(rad/s)/V$
Environment stiffness, K_e	$1200N/m$
Converting gain, K_p	$2.79 * 10^{-5} m/rad$
Operator stiffness, K_{op}	$500N/m$

Table 1: Constants of the system model.

where F_{op} is the force generated by the operator's muscles. M_{op}, B_{op}, and K_{op} denote mass, damper, and stiffness of the operator and the master arm respectively.

It should be noted that these operator's dynamic parameters may change during operation. For simplicity, the first two terms on the right hand side of (4) are omitted, and K_{op} is taken as constant.

The constants of the parameters of the tele-handshaking system is shown in Table1.

Since the master and slave are assumed to be physically equivalent, all parameters of both VIs for the master and slave should be the same. Each VI is represented by the following transfer function.

$$V(s) = \frac{1}{M_v s^2 + D_v s + K_v} \qquad (7)$$

3.3.4 Simulation

It is assumed that the master and the slave are physically equivalent. Thus the parameters of both VIs should be the same. An operator tried to pull the spring by moving the master arm, holding for a while, and then releasing it.

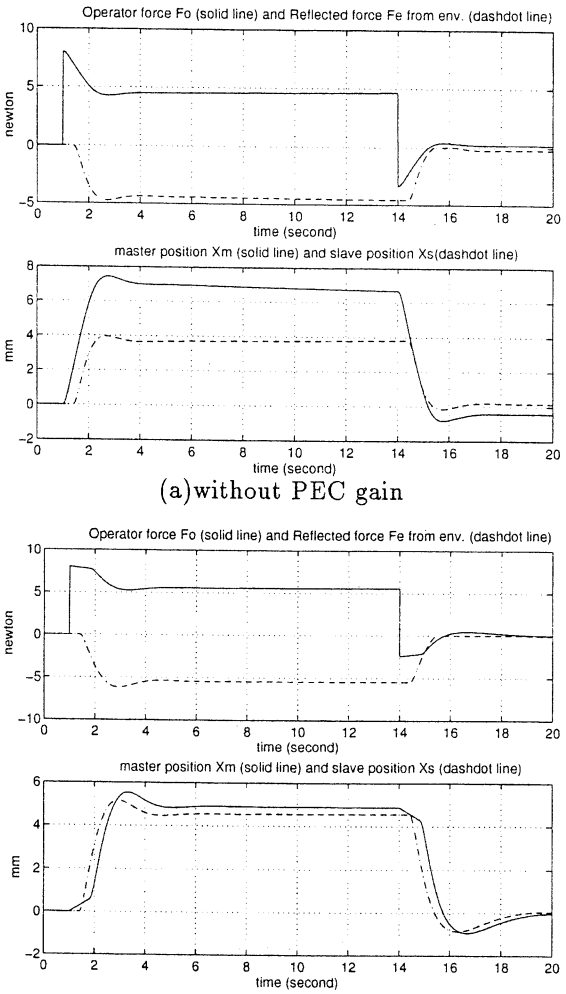
Similar to above experiment, simulation of the teleoperator with time delay was performed in two cases: 1) VI without PEC (PEC gain = 0) and 2) VIPEC.

In the simulation, the force exerted by an operator's muscles (F_{op}) was assumed to be a step function from $1^{s}t$ second to $14th$ second. The time delay of 400 ms was inserted in data transmission. The values of the parameters for simulation are shown in table 1. The parameters of both VIs, $M_v = 10kg$, $D_v = 1000N/(m/s)$, and $K_v = 20N/m$ are the same. PEC gain was set to 10.

Fig.11 shows the simulation results for the system with 400 ms of time delay. In both cases, with and without PEC, the system is stable and force sensation is realized. However, VIPEC succeeds to reduce the position error between the master and slave positions.

3.4 Experimental Result

The experimental results of the teleoperator with 400[ms] (Fig.12) of time delay agree well with the simulation results. The control approach, VIPEC, can re duce the position error between the master position and the slave position. However, as delay time increases,



(a)without PEC gain

(b) with PEC gain

Figure 11: Simulation Results of VIPEC with time delay (Time delay:400[ms])

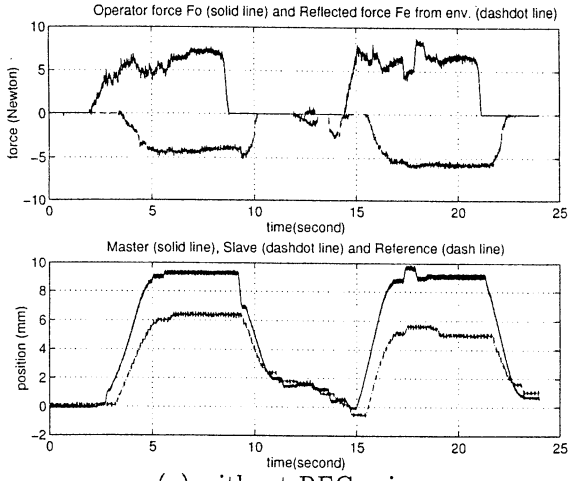
so does the position error between the master and slave. Here, the dynamics and the input forces to the both VIs are the same. Time delay (400[s]) is made in a computer.

Experimental data of the tele handshaking is shown in Fig.13. For simplicity to do implementation two HSDs and two host computers were installed in our laboratory. The data from two host computers was communicated through the Internet via a server workstation which was located at Osaka University. Each host computer sent the data to the server workstation every 120[ms]. The server forwarded the written data to the other host computer every 100[ms].

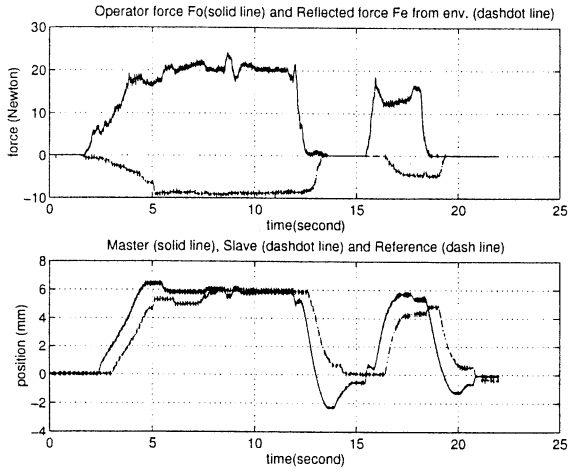
The data was composed of three float numbers: the force applied by the operator, the force reflected from environment, and the slave position. Therefore 12 bytes of data was transmitted through the Internet at each time.

The VIPEC was implemented on the HSD. The parameters of the VI were $M_v = 10[kg]$, $D_v = 800[N/(m/s)]$, and $K_v = 100[N/m]$. PEC gain was 2. VIPEC realized in the host computers ran at a 250 Hz interrupt rate.

Two operators succeeded to shake hands with each other and achieved force sensation through the HSD with VIPEC control approach. The position difference between the master and slave positions was small.



(a) without PEC gain



(b) with PEC gain

Figure 12: Experimental Result of Teleoperation

4 Conclusion

In this paper, we mentioned about two experimental systems of Robotic Network System(RNS) to show possibility of Computer Networked Robotics. The first system, our mobile robot: MUSEAR was controlled by an operator on remote hosts. An operator was watching a color video image transmitted through the Internet. In this experiment robot was playing as a physical agent from a information space.

The second system : HandShake Device (HSD) was shown. Two operators located on the different places could communicate with handshaking to each other. We could display force information, which was physical information through the network, to an operator by using network and robot technology. It means that a robot, especially, a haptic interface (a master system of teleoperation) as HSD is defined to display one of media such as a force. In that experiment, we used time-delay compensation with proposed VIPEC and shown good results of tele-handshaking between Tokyo and Osaka.

Above experiment systems had time-delay and variable time-delay which come out the instability of systems. We have to solve them as future works.

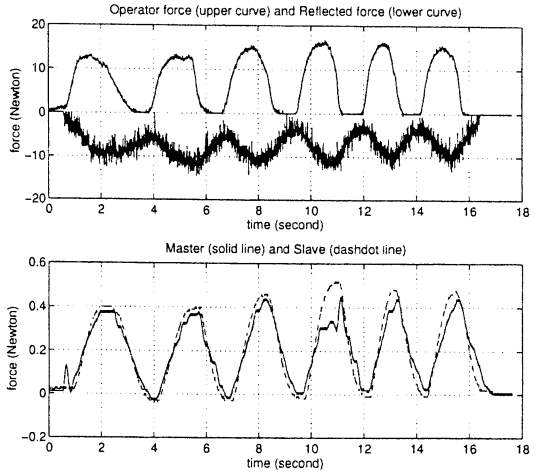


Figure 13: Tele-Handshake between Tokyo and Osaka

References

[1] E. Paulos and J. Canny: "Delivering Real Reality to the World Wide Web via Telerobotics," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1694-1699, 1996.

[2] K. Goldberg, M. Mascha, S. Gentner, e tal.: "Desktop Teleoperation via the World Wide Web," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 654-659, 1995.

[3] W. J. Book, H. Lane, L. J. Love, e tal.: "A Novel Teleoperated Long-Reach Manipulator Testbed and its Remote Capabilities via the Internet," *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1036-1041, 1996.

[4] Y. Kunii, H. Hashimoto: "Tele-Handshake Using HandShake Device", Proc. IECON' 95 , 1, 1995 pp. 179-182

[5] WITTENSTEIN Motion Control GmbH: "Roller screws", pp.5–7.

[6] S.Nojima, H.Hashimoto: "Master-Slave System With Force Feedback Based on Dynamics of Virtual Model", Proc. of ISMCR'94: Topical Workshop on Virtual Reality, pp.93–100, 1994.

[7] K. Furuta, K. Kosuge, Y. Shinote, H. Hatano: "Master-slave manipulator based on virtual internal model following control concept", Proc. IEEE Int. Conf. on Robotics and Automation, 1987, pp. 567-572

[8] M. Otsuka, N. Matsumoto, T. Idogaki, K. Kosuge, T. Itoh: "Bilateral telemanipulator system with communication time delay based on force-sum-driven virtual internal model", Proc. IEEE Int. Conf. on Robotics and Automation, 1995, pp. 344-350

[9] M.Suradech, H.Hashimoto: "Tele-Handshake Interface Based on Teleoperation with Time Delay", Proc. 7th Int. Power Electronics Motion Control Conference and Exhibition, 1996.(accepted)

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Fast Estimation of Motion Parameters for Vehicle-Camera Using Focus of Expansion

Zhencheng Hu, Keiichi Uchimura, and Shigeyasu Kawaji
Department of Computer Science, Faculty of Engineering, Kumamoto University
Kurokami 2-39-1, Kumamoto, 860-8555 Japan
Email: {ko, uchimura, kawaji}@eecs.kumamoto-u.ac.jp

Abstract: Estimation of motion parameters for vehicle-mounted camera plays a vital role in the research of autonomous vehicle's ego-state recognition and analysis of a dynamic scene. A fast motion parameters' estimation method with the introduction of camera's Focus of Expansion (FOE) is presented here. Compared with other approaches like using line matching of road lane-marks or tracking the horizon, proposed method purely uses points matching between the adjacent frames without need of any previous knowledge or recognition results. Unlike other feature matching algorithms, this method is able to determine camera's rotation and translation parameters by using only 3 pairs of matching points and thus more suitable for real-time application. Experiments on image sequences of real road scenes showed the effectiveness and precision of proposed method.

Keywords: Fast camera motion analysis, FOE, feature matching, intelligent vehicle, image sequence analysis

I. INTRODUCTION

In the research of autonomous robots and intelligent vehicles, more and more attention worldwide has been received on vision-based navigation systems in recent years[1][2]. As the basic and core part of these systems, especially for on-road running vehicles, the vision part is always designed to undertake following main tasks. 1) to detect and trace road lanes in a wide and near viewing area for keeping no deviation from the road, 2) to extract the 3-D road shape from both horizontal and vertical mapping in a tele-viewing area for the control of vehicle's speed and direction, 3) and to detect and avoid obstacles on the road.

All these detecting and measuring tasks absolutely depend on the precise state data of mounted camera, like pitch angle, elevation above road surface plane and heading angle relative to vehicle reference axis. Unfortunately, these state data are not stable while the vehicle is moving, especially on an uneven road or by a sudden acceleration or braking. Thus, the estimation of camera motion parameters has become a very important and indispensable technique for autonomous driving control and dynamic analysis of road scene.

Previous work in the area of camera motion analysis has been mainly in two categories: 1) with previous knowledge of road shape and vehicle locomotion data, 2) purely image fixation between adjacent frames. Methods developed in Refs.[1] and [2] estimate camera's motion parameters by using some previous recognition results like road lane-marks and horizon disappointing. These methods show some good performance in accuracy and efficiency because of their detailed analysis of road structure and measured vehicle locomotion, which is, however, computationally expensive and over-depended upon road features like lane-marks, and therefore lead to unsatisfied result when lane mark is covered by other vehicles or not exist at all. As for image fixation approaches like Eight-points algorithm, which is a well known tool for camera motion analysis, has often been criticized for being excessively sensitive to noise, and many refinements have been introduced like Ref.[3]. Optical flow is another well-used image fixation algorithm [4]. However by reason of its huge calculation cost and its difficulty for determining the accurate flow vectors, it is still unavailable to real applications.

Compared with these previous works, we present here a fast camera motion analysis method, which purely uses points matching between the adjacent frames. With the introduction of camera's Focus of Expansion (FOE), it is able to determine rotation and translation parameters theoretically by using only three pairs of matching points, which make it faster and more efficient for real-time applications.

In section II, we will interpret the camera motion analysis with the introduction of FOE theoretically. Then, the detailed image processing methods are proposed which include an FOE estimation method and a fast matching algorithm. In section IV, experimental results on real outdoor road image sequence show the effectiveness and precision of our approach.

II. CAMERA MOTION ANALYSIS WITH FOE

In the following analysis, we assume that the camera is mounted on an ordinary vehicle, which is driving on normal roads. Suppose the camera coordinate system (CCS) X_n-Y_n-Z_n and projection image coordinate system (ICS) x_n- y_n are fixed with respect to the camera as shown in Fig.1. Assume the center point of camera's lens is at the origin of the CCS, and the Z-axis is the optical axis of camera. Since the camera's optical axis is normally set vertical to the projection image, the x_n and y_n axis of ICS are parallel with the X_n and Y_n axis of CCS respectively. The vehicle's reference axis is labeled as ***OL***.

A. Camera motion analysis equations

The motion of mounted camera is always treated as the general motion of rigid bodies, and thus can be decomposed into a translation along three independent spatial axes and a rotation around those axes. As shown

in Fig.1, this six-degree-of-freedom motion can be represented by translation vector $\boldsymbol{T}(T_X,T_Y,T_Z)$ and rotation vector $\boldsymbol{\Omega}(\Omega_X,\Omega_Y,\Omega_Z)$ with respect to the X,Y and Z axis of CCS. Suppose any static point in space, say P, the changes of its CCS coordinate by reason of camera's motion $\boldsymbol{T}$ and $\boldsymbol{\Omega}$, can be treated as the coordinate changes caused by its relative motion $\boldsymbol{T}'(-T_X,-T_Y,-T_Z)$ and $\boldsymbol{\Omega}'(-\Omega_X,-\Omega_Y,-\Omega_Z)$ towards a relative static CCS.

If the 3-D position of point P at time t_n is (X_n,Y_n,Z_n), we can calculate its new position after its relative motion at time t_{n+1} by:

$$\begin{pmatrix} X_{n+1} \\ Y_{n+1} \\ Z_{n+1} \end{pmatrix} = \boldsymbol{R} \begin{pmatrix} X_n \\ Y_n \\ Z_n \end{pmatrix} + \boldsymbol{T}' \tag{1}$$

where $\boldsymbol{R}$ is a 3x3 orthonormal matrix representing the rotation. For small roll angle Ω_X,Ω_Y and Ω_Z, the approximations $\cos\theta \approx 1$ and $\sin\theta \approx \theta$ hold, thus R is always calculated by

$$\boldsymbol{R} = \begin{pmatrix} 1 & \Omega_Z & -\Omega_Y \\ -\Omega_Z & 1 & \Omega_X \\ \Omega_Y & -\Omega_X & 1 \end{pmatrix} \tag{2}$$

And $\boldsymbol{T}'$ is the relative translation vector.

$$\boldsymbol{T}' = (-T_x,-T_y,-T_z)^T \tag{3}$$

Therefore, equation (1) can be rewritten as follows.

$$\begin{cases} \Delta X_n = X_{n+1} - X_n = \Omega_Z Y_n - \Omega_Y Z_n - T_X \\ \Delta Y_n = Y_{n+1} - Y_n = \Omega_X Z_n - \Omega_Z X_n - T_Y \\ \Delta Z_n = Z_{n+1} - Z_n = \Omega_Y X_n - \Omega_X Y_n - T_Z \end{cases} \tag{4}$$

B. *Introduction of FOE*

The intersection of the 3-D vector representing camera's translation direction and the projection plane is called the focus of expansion (FOE)[5]. As shown in Fig.2, the FOE is vehicle relative axis $\boldsymbol{OL}$'s intersection on the projection image. The FOE plays a vital role in the analysis of camera motion. Once the FOE has been determined, we can estimate distances from the camera to points in the scene being imaged. While there is an ambiguity in scale, it is possible to calculate the ratio of distance to speed, and therefore can estimate Time-To-Impact (TTI) between the camera and objects in the scene. Also, there is an important feature of FOE that all the motion vectors on the projection image intersect at the FOE if there is only translation motion concerned between the camera and its environment.

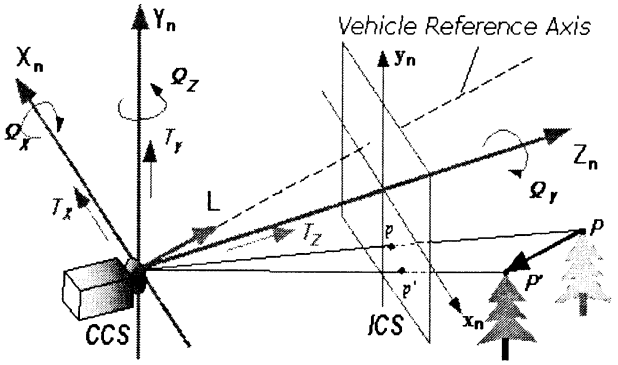


Figure 1. Relationship between CCS and ICS

For vehicle-mounted case, since the camera and the vehicle are mounted together (it means the relative location between ICS and CCS is stable), and vehicle's translation motion is only in the direction of vehicle relative axis, we can obtain a conclusion that the camera translation direction is stable in camera coordinate system. And therefore once the camera is mounted, the coordinate of the FOE in ICS is stable.

Suppose FOE is at (x_0, y_0) in ICS, then the following equation can be obtained through Fig.2.

$$\frac{x_0}{T_X} = \frac{y_0}{T_Y} = \frac{f}{T_Z} \tag{5}$$

where f is camera's focus distance.

Therefore,

$$x_0 = f\frac{T_X}{T_Z}, \quad y_0 = f\frac{T_Y}{T_Z} \tag{6}$$

C. *Camera Motion Parameters Estimation Using FOE*

If we suppose the projection location on image of point P is at $p(x_n, y_n)$ (see Fig.1), a following perspective transform can be used to obtain its ICS coordinate.

$$x_n = f\frac{X_n}{Z_n}, \quad y_n = f\frac{Y_n}{Z_n} \tag{7}$$

The instantaneous motion velocity (u, v) of p on the image plane can be given by the time differential of (7).

$$\begin{aligned} u = \dot{x}_n &= \frac{d}{dt}(f\frac{X_n}{Z_n}) = f(\frac{\dot{X}_n}{Z_n} - X_n\frac{\dot{Z}_n}{{Z_n}^2}) \\ v = \dot{y}_n &= \frac{d}{dt}(f\frac{Y_n}{Z_n}) = f(\frac{\dot{Y}_n}{Z_n} - Y_n\frac{\dot{Z}_n}{{Z_n}^2}) \end{aligned} \tag{8}$$

If the sampling time between two adjacent frames is short enough, we can use the following equations to approximate moving distance of projection point p.

$$\begin{cases} x_{n+1} - x_n = uT = f(\frac{\dot{X}_n}{Z_n}T - X_n\frac{\dot{Z}_n}{{Z_n}^2}T) = f(\frac{\Delta X_n}{Z_n} - X_n\frac{\Delta Z_n}{{Z_n}^2}) \\ y_{n+1} - y_n = vT = f(\frac{\dot{Y}_n}{Z_n}T - Y_n\frac{\dot{Z}_n}{{Z_n}^2}T) = f(\frac{\Delta Y_n}{Z_n} - Y_n\frac{\Delta Z_n}{{Z_n}^2}) \end{cases} \tag{9}$$

With (4) and (5), it gives

$$\begin{cases} x_{n+1} - x_n = \frac{x_n y_n}{f}\Omega_X - \frac{f^2 + {x_n}^2}{f}\Omega_Y + y_n\Omega_Z + \frac{T_Z}{Z}(x_n - x_0) \\ y_{n+1} - y_n = \frac{f^2 + {y_n}^2}{f}\Omega_X - \frac{x_n y_n}{f}\Omega_Y - x_n\Omega_Z + \frac{T_Z}{Z}(y_n - y_0) \end{cases} \tag{10}$$

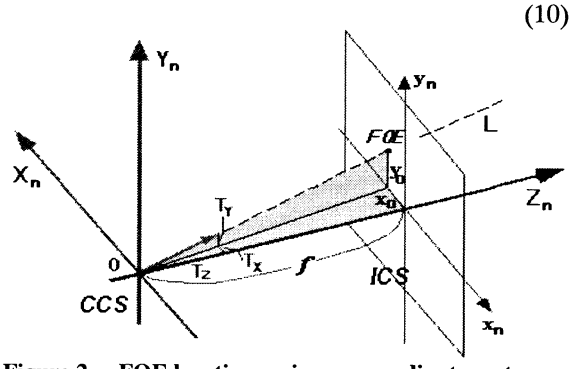


Figure 2. FOE location on image coordinate system

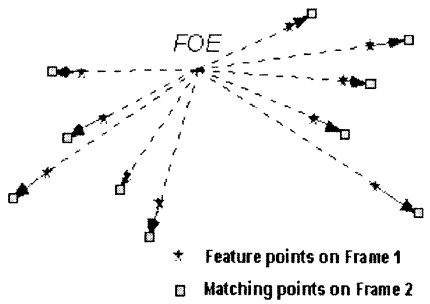


Figure 3. Matching pairs and FOE

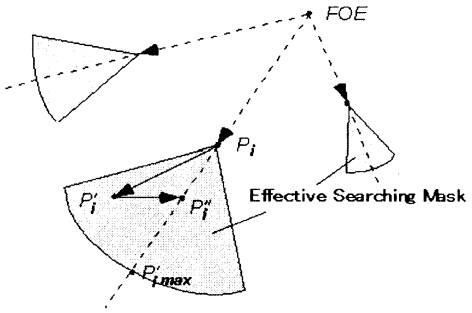


Figure 4. Effective searching mask for points matching

If we assume $T_Z/Z \neq 0$, we can obtain

$$\frac{(x_{n+1}-x_n)-(\frac{x_n y_n}{f}\Omega_X-\frac{f^2+x_n^2}{f}\Omega_Y+y_n\Omega_Z)}{(y_{n+1}-y_n)-(\frac{f^2+y_n^2}{f}\Omega_X-\frac{x_n y_n}{f}\Omega_Y-x_n\Omega_Z)}=\frac{x_n-x_0}{y_n-y_0} \tag{11}$$

We observe that, with these identical transformations and the introduction of FOE, camera's complicated motion, which includes rotation and translation six-degree-of-freedom as mentioned in part A of this section, can be represented by a simple equation (11) with only three rotation parameters unknown $(\Omega_X,\Omega_Y,\Omega_Z)$. For each pair of matching points between two adjacent frames one constraint can be provided. Therefore, only three pairs of matching points theoretically can obtain these rotation parameters. Other translation parameters or the relative distance can be obtained using (5) and (10).

III. IMAGE SEQUENCE PROCESSING METHODS

In this section, a layout of image processing steps is shown below, and more details are given in our previous works[8].

A. Estimation of FOE

Many works have been done upon the estimation of camera's FOE. From these studies, we refine Lowton's algorithm [6], which is based on Course to Fine method to reduce searching times.

As discussed in Part B of Section II, when there is only translation motion between the camera and its environment, if we composite both feature point and its matching one into one image, all the lines connecting corresponding pairs will intersect at the FOE (see Fig.3). Suppose the feature points in frame F_t, which can be extracted by using 3-stage angle controllable corner extraction method [7], are at P_i $(i=1,2,...,k)$, we can say that the position of its matching point will be on the extended line connecting FOE and P_i. Assuming vehicle's moving on straight road can be approximately treated as pure translation motion, the FOE can thus be estimated using straight road scene image sequence based on the above concept [8].

B. Feature matching of image sequence

For pure translation motion case, the matching point is on the radiant lines started from the FOE and thus the searching area is only on a line segment for each feature point. If the rotation motion is also been taken into account, the matching will become more complicate. Let us suppose the moving distances with respect to rotation as x_Ω on x-axis, and y_Ω on y-axis of ICS. Therefore we can rewrite (11) as following:

$$\frac{x_{n+1}-x_\Omega-x_n}{y_{n+1}-y_\Omega-y_n}=\frac{x_n-x_0}{y_n-y_0} \tag{12}$$

where,

$$\begin{cases} x_\Omega=\dfrac{x_n y_n}{f}\Omega_X-\dfrac{f^2+x_n^2}{f}\Omega_Y+y_n\Omega_Z \\ y_\Omega=\dfrac{f^2+y_n^2}{f}\Omega_X-\dfrac{x_n y_n}{f}\Omega_Y-x_n\Omega_Z \end{cases} \tag{13}$$

From (13) we can see that even for the complicated case (translation + rotation), if the rotation effect x_Ω and y_Ω are eliminated from the real matching position P_i' (x_{n+1}, y_{n+1}), the new point P_i'' $(x_{n+1}-x_\Omega, y_{n+1}-y_\Omega)$ will still be on the extended line connecting FOE and P_i (see Fig.4).

Therefore, we propose here an effective searching mask for matching point P_i'. Unlike the usual square or rectangle searching window, it is a fan shape area that starts from P_i and the symmetrical axis is the extended line connecting FOE and P_i. The radius is determined by the farthest possible position $P_{i\,max}'$, and spread angle is determined by camera possible rotation angles around X, Y and Z-axis of CCS.

Our effective searching mask does not only reduce the computational time but also provide more accurate matching results since the mask is defined more accurately in consideration of camera's motion feature. The SSDA method is used for the computation of maximum likelihood.

C. Estimation of camera motion parameters

With these matching pairs, camera motion parameters can be calculated by using proposed method as discussed in Section II. We use the least-squares method to reduce noise, digital error and mismatching caused by feature points from other moving objects.

IV. EXPERIMENTAL RESULTS

The real road images we used were obtained by a 1/2-inch color CCD camera that was mounted on an ordinary front wheel steering passenger car. The valid frame size is 256x256, the focus distance is equal to 480 pixels and the sampling time is 1/30 second. The experimental condition is in heavy cloudy weather and test roads include expressway and countryside normal roads as shown in Fig.5.

1) *Feature matching using FOE:* The feature point matching method has been tested in different road conditions. Figure 6 shows some example results of straight and curve roads. It is shown that the proposed method based on FOE gives very accurate matching results, and the mismatching rate is almost 0.
2) *Motion parameter estimation on highway road:* In general, when the vehicle is driving on a flat straight road, camera's rotation around each axis is extremely slight. For a flat curve road, the rotation around Y-axis dominates, and rotation around X-axis is extremely slight. Test results of a straight part and a left curve of expressway (see Fig.5) are shown in Fig.7 and Fig.8. For the case of straight road, estimation results of rotation parameters Ω_X and Ω_Y change only in a small range of $\pm 0.5 degrees$, and it is corresponding with the theoretical analysis. As shown in Fig.8, estimation results of Ω_Y change around $+2.1 degrees$ and Ω_X is almost $0 degrees$. This is also verified by the real experiment situation of vehicle's turning left on a left curve with almost the same speed.

V. CONCLUSIONS

A fast dynamic estimation method of motion parameters for vehicle-mounted camera using FOE has been presented. Because of the FOE position's stability on a vehicle-mounted camera, we can directly use FOE for feature matching, and then estimate camera's motion parameters using by simple computations. Experiments on image sequences of real general road scenes showed the effectiveness and the precision of our approach.

Main advantages of the proposed approach can be given as follows. (1) All the camera's motion parameters can be accurately estimated theoretically using only 3-pairs of matching points. It makes this approach less computation cost and more suitable for real time applications. (2) The effective searching mask proposed here does not only reduce the computational time but also provide more accurate matching results since the mask is defined more accurately in consideration of camera's motion feature. (3) This method extracts the feature points from all image pixels without need of any previous knowledge of recognition result, which make it more efficient and flexible.

REFERENCE

[1] Dickmanns, E.D., "Vehicles Capable of Dynamic Vision", IJCAI97, pp.1577-1592 (1997)
[2] Nohsoh K., Ozawa S., "A Simultaneous Estimation of Road Structure and Camera Position from Continuous Road Images ", IEICE Trans. D-II, No.4, 764-773, 1994 (in Japanese)
[3] Hartley, R.I, "In Defense of Eight-Point Algorithm", IEEE Trans. PAMI, Vol. 19, No. 6, pp.580-593 (1997)
[4] Hummel,R.A., Sundareswaran,V., "Motion Parameter Estimation from Global Flow Field Data", IEEE Trans. PAMI, Vol.15, No.5, pp.459-476 (1993)
[5] Jain, R.C., "Direct Computation of the Focus of Expansion", IEEE Trans. PAMI, Vol.5, No.1, pp.58-64 (1983)
[6] Lowton, D., "Precessing Dynamic Image Sequence from Moving Sensor", Ph.D Thesis, TR-84-05, Univ. of Massachusetts (1985)
[7] Hu Z., Uchimura K., "Recognition of Horizontal Shape Models for General Roads", IEICE Trans. A, Vol. J81-A, No.4, pp.580-589, 1998 (in Japanese)
[8] Uchimura K., Hu Z., "Lane Detection and Tracking Using Estimated Camera Parameters for Intelligent Vehicles", 1998 IEEE Intel. Conf. on Intelligent Vehicles, pp.11-16, Germany (1998)

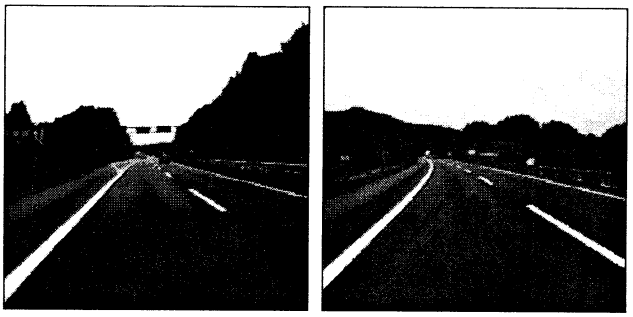
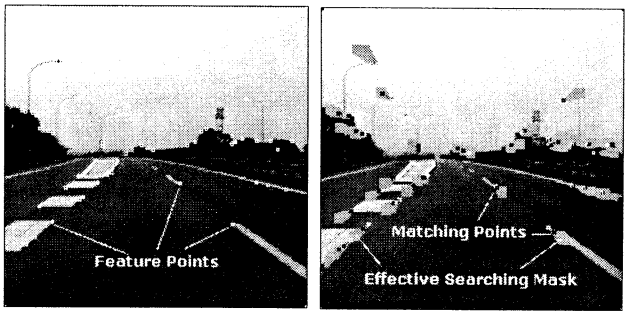
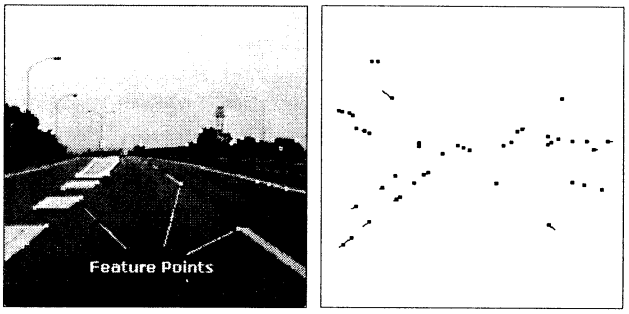


Figure 5. Testing real roads



(a) Feature points in frame#1 **(b) Points matching**



(c) Matching results in frame#2 **(d) Motion vectors**

Figure 6. Feture matching using FOE

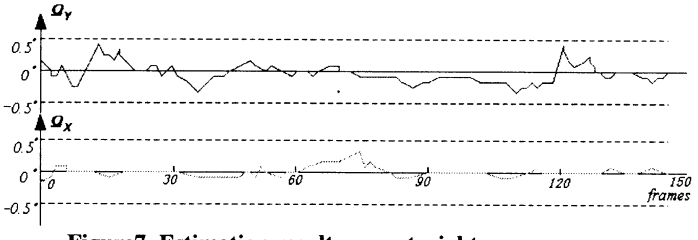


Figure7. Estimation results on a straight expressway

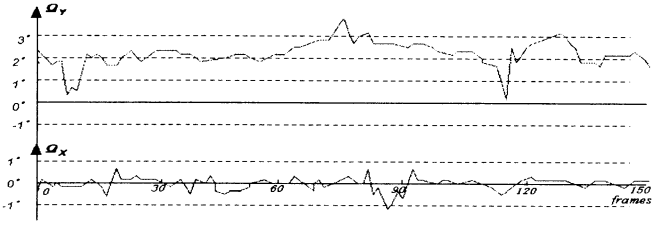


Figure 8. Estimation results on a left curve expressway

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Intelligent Control of Robots in Coordination

Kazuhiro Kosuge*, Yasuhisa Hirata*, Koji Takeo*
Hajime Asama**, Hayato Kaetsu** and Kuniaki Kawabata**

*Department of Machine Intelligence and Systems Engineering,
Tohoku University
Aoba-yama01, Sendai 980-8579, JAPAN
**Biochemical Systems Laboratory,
The Institute of Physical and Chemical Research, RIKEN
Hirosawa 2-1, Wako-shi, Saitama, 351-0198, JAPAN

Abstract

In this paper, we consider a problem of handling an object by multiple robots in coordination. First, we propose a decentralized control algorithm for this problem. One of the robots is referred to the leader and motion command of the object is given only to the leader. Each of the rest of the robots referred to a follower estimates the motion of the object commanded to the leader by itself without any dense communication. Several experimental results will illustrate the system.

1 Introduction

When we would like to transport a large and heavy object, we carry it in cooperation with other people. To utilize multiple robots in coordination is a natural extension of such human behavior to the robots. In this article, we propose an intelligent decentralized control algorithm of multiple mobile robots handling an object in coordination like humans.

Much research has been already done for the motion control of multiple manipulators handling an object in coordination [1]-[5]*etc* . However, most of the control algorithms proposed so far have been designed for the centralized control system, that is, a single controller is supposed to control all of the robots in a centralized way. The centralized control system could handle two or three manipulators in coordination, but it is not easy to control many mobile robots in coordination because of its implementation problem and computational burden.

In this article, we develop a decentralized control algorithm for handling a single object by multiple robots in coordination like humans. The algorithm could be applied to multiple mobile manipulators in coordination. In the proposed control system, the motion command of the object is given to one of the robots, referred to as a leader. The other robots are referred to as followers and each follower, which is controlled by its own controller, estimates the motion of the leader. The proposed control algorithm is experimentally applied to omnidirectional mobile robots first, and the results illustrate the validity of the proposed control algorithm.

2 Principle of Coordination

2.1 Two Robots in Coordination

In this section, we explain the decentralized control algorithm. For the simplicity of explanation, we first consider the case of two autonomous omnidirectional mobile robots, each of which has three degrees of freedom of motion. We assume that each robot is controlled by its own controller in a decentralized way. The desired trajectory of the object is given to the leader and the follower estimates the desired motion of the object commanded to the leader to transport the object in coordination with the leader.

We assume that each robot has the following dynamics by an appropriate controller as shown in Fig.1.

$$D\Delta\dot{x}_l + K\Delta x_l = f_l - {f_l}^{in} \quad (1)$$

$$D\Delta\dot{x}_1 + K\Delta x_1 = f_1 - {f_1}^{in} \quad (2)$$

where $D \in R^{3\times3}$ and $K \in R^{3\times3}$ are positive damping and stiffness matrices, $f_l, f_1 \in R^3$ be a forces

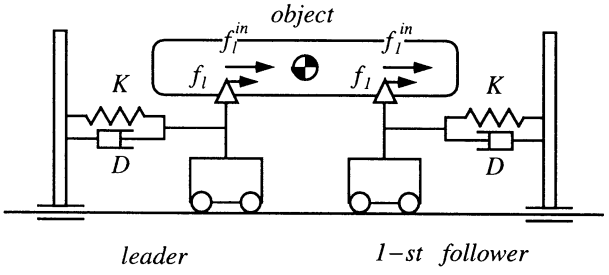


Figure 1: Compliant Motion of Robot

applied to the robots, $f_l^{in}, f_1^{in} \in R^3$ be a specified internal forces applied to the object by the robots, and $\Delta x_l, \Delta x_1 \in R^3$ be the trajectory deviations of the robots according to the forces applied to the robots. The subscripts l and 1 indicate that the variables with the subscripts are related to the leader and the follower respectively.

Let $x_d, x_{e1} \in R^3$ be the desired trajectories of the leader and the follower respectively and $x \in R^3$ be the real trajectory of the object. Under the assumption that each robot holds the object firmly and no relative motion between the object and each robot occurs, these deviations, Δx_l and Δx_1, are expressed as follows;

$$\Delta x_l = x - x_d \tag{3}$$

$$\Delta x_1 = x - x_{e1} \tag{4}$$

Under the assumption that the external force applied to the object is negligible, the relationship between Δx_l and Δx_1 is expressed as follows from Eq.(1) and (2);

$$\Delta x_l + \Delta x_1 = 0 \tag{5}$$

Eliminating x from Eq.(3) and (4), we obtain

$$\Delta x_1 - \Delta x_l = x_d - x_{e1} \tag{6}$$

Let Δx_{d1} be the trajectory estimation error of the follower, then, Δx_{d1} is expressed by

$$\Delta x_{d1} = x_d - x_{e1} \tag{7}$$

From Eq.(5), (6) and (7), Δx_{d1} can be written as follows;

$$\Delta x_{d1} = 2\Delta x_1 \tag{8}$$

It should be noted that the follower can calculate Δx_{d1} using the observable variable Δx_1.

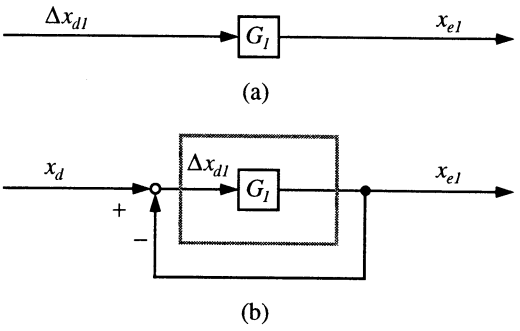


Figure 2: Estimator

Let us consider how x_d is estimated using Δx_{d1}. Let G_1 be the transfer function, which estimates x_d, as x_{e1}, based on Δx_{d1} as shown in Fig.2(a). From Eq.(7), Fig.2(a) can be rewritten as a feedback system shown in Fig.2(b). To eliminate the steady-state position and velocity estimation errors, the transfer function G_1 is designed as follows;

$$G_1 = \frac{a_1 s + b_1}{s^2} \tag{9}$$

2.2 n+1 Robots in Coordination

We extend the result in the previous section to a general case. It is impossible for the i-th follower to estimate the desired trajectory of the leader because the trajectory deviation of the i-th follower, which was used for the estimation of the desired trajectory of the leader in previous case, is affected by motions of all of the robots. Therefore, for the i-th follower, the robots are classified into two groups; one is the i-th follower itself and the other is the rest of the robots including the leader. In this paper, we refer to the rest of the robots as the i-th virtual leader. The i-th virtual leader consists of the leader, and j-th followers ($j = 1, \ldots, i-1, i+1, \ldots, n$) as shown in Fig.3. For the i-th follower, the i-th virtual leader behaves as if it were a real leader. Using the concept of the virtual leader, the i-th follower estimates the desired trajectory of the i-th virtual leader based on the estimation algorithm when two robots handle a single object in coordination.

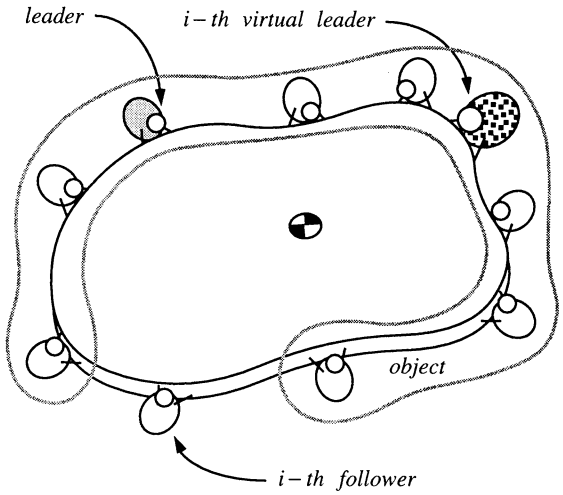


Figure 3: Virtual Leader

3 Experimental Results

We did experiments using two autonomous omnidirectional mobile robots, ZEN, developed by RIKEN [8]. Each mobile robot has three degrees of freedom of motion and equipped with the Body Force Sensor[9] as shown in Fig.4. The control algorithm explained in the previous section was implemented using VxWorks. The sampling rate was 1024Hz.

In this experiment, the leader is given a desired trajectory along y-axis which is calculated by a fifth order function. The orientations of all of the robots are kept constant during the transportation of the object. The constant internal force is also specified. The results are shown in Fig. 5. The solid lines show the results relating to the leader. The dotted lines show the results relating to the follower. You can see that the transportation of a single object has been successfully achieved.

Fig. 6 shows an example of the experiments.

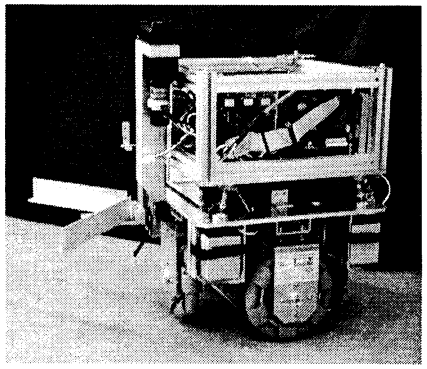
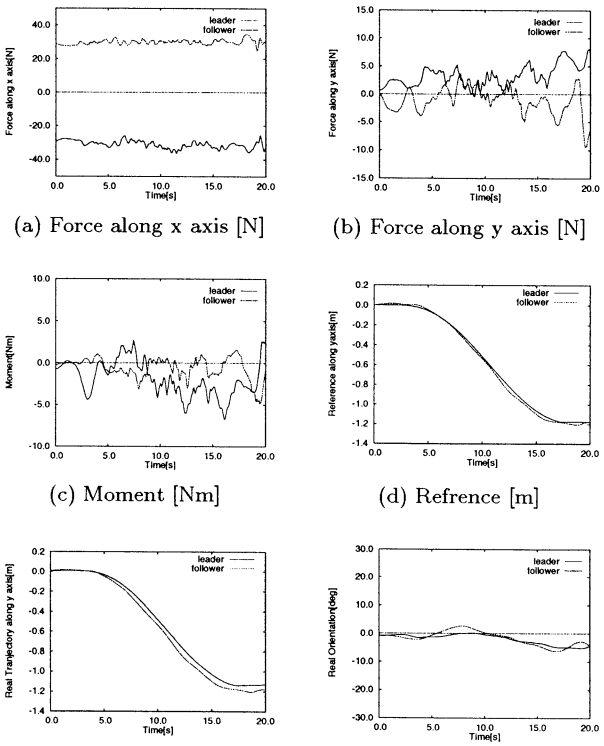


Figure 4: Experimental System



(a) Force along x axis [N]
(b) Force along y axis [N]
(c) Moment [Nm]
(d) Refrence [m]
(e) Real Trajectory y axis [m]
(f) Real Orientation [deg]

Figure 5: Experimental Results

4 Conclusions

In this paper, we proposed a decentralized motion control algorithm of multiple robots in coordination. The algorithm realizes a natural robot behavior for handling a single object in coordination. The robots handle an object in coordination although only the leader knows how to move. The collective behavior was realized through the estimation of leader's motion by each follower based on information obtained at each follower. Experimental results illustrated the concept of the system.

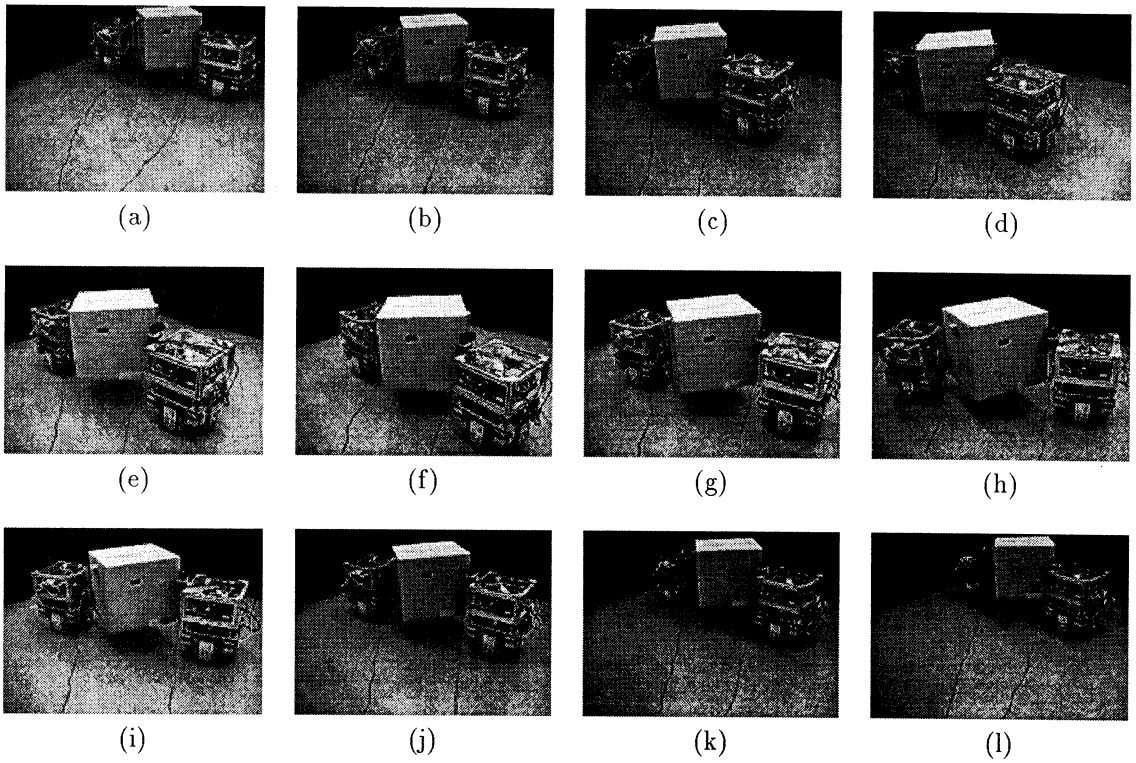


Figure 6: Example of experiment

References

[1] E.Nakano, S.Ozaki, T.Ishida, I.Kato, "Cooperational Control of the Anthropomorphous Manipulator "MELARM"", *Proc. of 4th International Symposium on Industrial Robots*, pp.251-260, Tokyo 1974.

[2] Y.F. Zheng, J.Y.S. Luh, "Optimal Load Distribution for Two Industrial Robots Handling a Single Object", *Proc. of IEEE International Conference on Robotics and Automation*, pp.344-349, 1988.

[3] Y. Nakamura, K. Nagai, T. Yoshikawa, "Mechanics of Cooperative Manipulation by Multiple Robotic Mechanism", *Proc. of 1987 International Conference on Robotics and Automation*, pp.991-998, 1987.

[4] K. Kosuge, J. Isikawa, "Task-Oriented Control of Single-Master Multi-Slave Manipulator System", *Robotics and Autonomous Systems*, Vol. 12, pp.95–105, 1994.

[5] Daniel J. Stillwell and John S. Bay, "Toward the Development of a Material Transport System using Swarms of Ant-like Robots", *Proc. of 1993 IEEE Int. Conf. on Robotics and Automation*, pp.318–323, 1993.

[6] J. Ota, Y. Buei, T. Arai, H. Osumi, K. Suyama, "Transferring Control by Cooperation of Two Mobile Robots", *Journal of the Robots Society of Japan*, Vol. 14, pp.263–270, 1996.(In Japanese)

[7] K. Kosuge, T. Oosumi, "Decentralized Control of Multiple Robots Handling an Object", *Proc. of 1996 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp.318–323, 1996.

[8] H. Asama, M. Sato, H. kaetsu, K. Ozaki, A. Matsumoto, I.Endo, "Development of an Omni-Directional Mobile Robot with 3 DoF Decoupling Drive Mechanism" *Journal of the Robots Society of Japan*, Vol. 14, pp.249–254, 1996.(In Japanese)

[9] K. Kosuge, T. Oosumi, Y. Hirata, H. Asama, H. Kaetsu, K. Kwabata "Handling of a Single Object by Multiple Autonomous Mobile Robots in Coordination with Body Force Sensor", *Proc. of 1998 IEEE Int. Conf. on Intelligent Robots and Systems*, pp.1419–1424, 1998.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Study on Coalition Formation Process in the Iterated Multiple Lake Game

Tomohisa YAMASHITA Keiji SUZUKI Azuma OHUCI
Research Group of Complex Systems Engineering
Graduate School of Engineering, Hokkaido University
Kita 13, Nishi 8, Kita-ku, Sapporo, 060-8628 JAPAN

Abstract

In this paper, we extend the environment management game, "The Lake", to the agent based iterated game to discuss more dynamic coalition formative process. In extended game, many factories often become free-riders to increase its own payoff, and then they fail to form desirable coalition and acquire low payoff. Our purpose is to prevent factories from becoming free-riders, to make them form a desirable coalition, and to increase the payoff of player in such situation. Here, we introduce the strategy of local government that it gives factory incentive. We compare two cases with incentive and without of it, and then analyze the effect of incentive.

1 Introduction

In this paper, we propose the agent based iterated game of coalition formation for discussing what kinds of local interactions of agents producing complex formation process. In agent based game simulations, while cooperative or competitive relations between agents have been developed, like as prisoners' dilemmas and market game [3], coalition based game simulations are not paid attention until now.

To develop the coalition formative game based on the agents, we extend "The Lake" [2] that is known as one of the environment management games and the social dilemmas. In the original game, it is hard to make the whole coalition and prevent the factories becoming the free-riders. Therefore, we introduce the local governments as new players and multiple lakes for the factories selecting the operating place [1]. When we introduce local government with the strategy levying a tax and imposing a penalty, the payoff from being free-riders decreases, and then factories tend to form treating coalition. Although government has this strategy, treating coalition is not formed often and the payoff of factory is low. Here, we introduce the strategy of local government that it gives treating coalition a subsidy as incentive.

We pay attention to whether a subsidy leads factories to form treating coalition. In this paper, we compare the model with incentive and the model without incentive, and observe how subsidy as incentive effect formation of treating coalition and the payoff of factory.

2 Agent based Multiple Lake Game with Local Governments

Outline of agent based Iterated Game

We extend "The Lake" [2] to develop the coalition formative game based on the agents. In order to prevent the factories becoming the free-riders, "local government" as new player is introduced in this game [1]. The local government wish to levy a tax on all factories around the lake, imposes a penalty on the anti-treating coalition and to give subsidy as incentive to treating coalition. We multiply the lake to introduce the dilemma between the local governments assigned into the lakes.

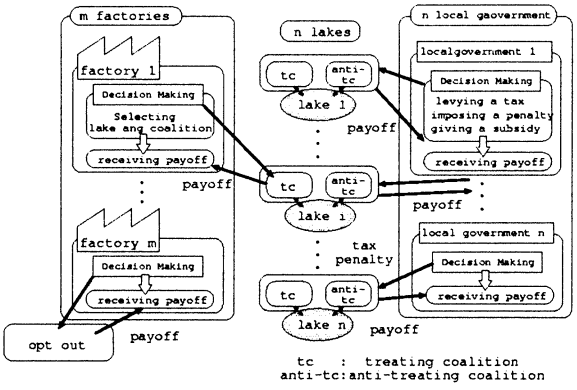


Figure 1: Outline of Agent based Multiple Lake Game

The decision making process will repeat until constant times, T. After the process, the evolving phase is applied to find the adaptive strategies for the agents

by evolutionary operators based on the received payoff. Here, one generation in this game is defined as a set of the decision making process and the followed evolving phase. In the iteration of the generations, K, we will expect the agents to produce the adaptive coalition formations based on their local interactions.

In this game model, the number of the lake is set as n. We define the coalition set as follows.

$$\{\{S_i, \overline{S_i}\}, opt-out\} \quad i = 1, ..., n \tag{1}$$

S_i :the treating coalition of lake i
$\overline{S_i}$:the anti-treating coalition of lake i

Pollution rate of lake

We assume that the natural resources of the lakes are limited to form more realistic game. Above mentioned the cost of treating its own water in the lake raises according to the pollution rate going up. In this model, we formulate the pollution rate with the following four elements.

$\overline{|S_i(k)|}$:the average size of a tc in k generation
$\overline{|\overline{S_i}(k)|}$:the average size of an $anti$-tc in k generation
L_1 :the threshold concerning a limited size of an $anti$-tc
L_2 :the threshold concerning a limited sum of factories

The pollution rate is fixed during one generation and renewed based on these parameters before the next generation.

The strategy and the payoff function of local government

The purposes of the local governments are to maximize their revenues. The strategies of the local governments change the tax and penalty with -1 (decreasing), ± 0 and +1 (increasing) from the current value. At t times game in k generation, the strategy set for current tax $\Omega_{T_i}(k,t)$ and penalty Ω_{Pen_i} are represented as

$$\Omega_{T_i}(k,t) = \{T_i(k,t-1) - 1, T_i(k,t-1), T_i(k,t-1) + 1\} \tag{2}$$

$$\Omega_{Pen_i}(k,t) = \{Pen_i(k,t-1) - 1, Pen_i(k,t-1), Pen_i(k,t-1) + 1\} \tag{3}$$

The strategy of the local governments determines the subsidy from the three values. The subsidy is paid from the sum of revenue till current game. The strategy set for subsidy $\Omega_{Sub_i}(k,t)$ is represented as

$$\Omega_{Sub_i}(k,t) = \{v(\overline{S_i})/|\overline{S_i}| - v(S_i)/|S_i|, const_{subsidy}, 0\} \tag{4}$$

$const_{subsidy}$:canstant value

The payoff function (revenue) of the local government is defined as follows;

$$g(T_i(k,t), Pen_i(k,t)) = (|S_i| + |\overline{S_i}|)T_i(k,t) + |\overline{S_i}|Pen_i(k,t) \tag{5}$$

$|S_i|$:the size of treating coalition of lake i
$|\overline{S_i}|$:the size of anti-treating coalition of lake i

The strategy and a characteristic function of factory

The strategies of the factories, to maximize the payoff (to minimize the cost for treating and purifying the water), are to select the operating places from the lakes and to decide which coalition they should participate in. "*Opt-out*" as additional strategy for the factories is introduced. Choosing this strategy means that not operating around lakes.

The strategy set of factory j is represent as

$$\Pi_j = \{\{S_i, \overline{S_i}\}, opt-out\} \quad i = 1, ..., n \tag{6}$$

The characteristic function of S_i and $\overline{S_i}$ can be constructed as follows;

$v(S_i) =$

$$\begin{cases} \alpha - |S_i|(|\overline{S_i}| + \beta)Po_i(k)D - T_i(k,t) - Pen_i(k,t) + Sub_i(k,t) \\ \quad if \quad |S_i|/|\overline{S_i}| < B/D \\ \alpha - |\overline{S_i}|(B - |S_i|D)Po_i(k) - T_i(k,t) \\ \quad otherwise \end{cases} \tag{7}$$

$$v(\overline{S_i}) = \begin{cases} \alpha - |S_i|Po_i(k)D - T_i(k,t) - Pen_i(k,t) \\ \quad if \quad |S_i|/|\overline{S_i}| < B/D \\ \alpha - (|S_{2i}|)^2 Po_i(k)D - T_i(k,t) - Pen_i(k,t) \\ \quad otherwise \end{cases} \tag{8}$$

As sharing method of the coalition value to the members, the payoff of the factories is defined by the coalition value divided with the number of the members. The reason is that all factory is symmetric in the coalitions. Therefore, the payoff function of the factory is defined as follows;

$f_j(\pi_j, \pi_{-j}) =$

$$\begin{cases} v(S)/|S| \quad if \quad select \quad coalition \\ const \quad if \quad select \quad the \quad strategy \quad to \quad opt \quad out \end{cases} \tag{9}$$

π_j :the strategy of the factory j at t times game in k generation,π_j(k,t)
π_{-j} :the vector of strategies except factory j $(\pi_1, ..., \pi_{j-1}, \pi_{j+1}, ..., \pi_m)$
S :the coalition that the factory participating in

Evaluation of strategy

In this game, the players make decisions on the basis of the continuously updated values, "expected payoffs", that relate to the reactive decisions in the iterated game and the evolution of the strategies.

Reactive Strategy with FSM

In order to determine the next decision based on the expected payoff, we investigate the finite state machines (FSMs) like as the iterated prisoner's dilemma [3].

Evolution of Strategy with GA

In order to acquire the adaptive strategy, the genetic algorithm (GA) is applied to the FSMs in each agent.

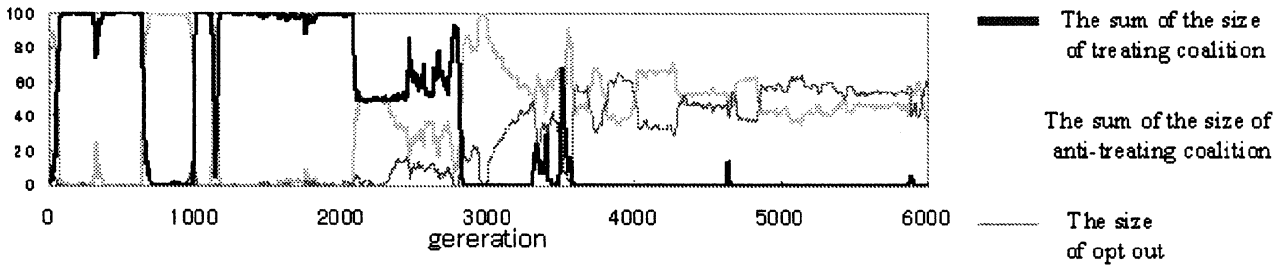


Figure 2: The size of *tc*,*anti-tc* and *opt-out* before the introduce of incentive

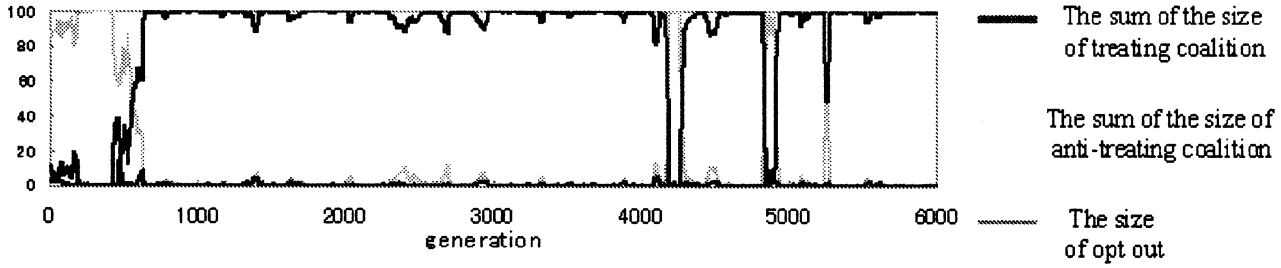


Figure 3: The size of *tc*,*anti-tc* and *opt-out* after the introduce of incentive

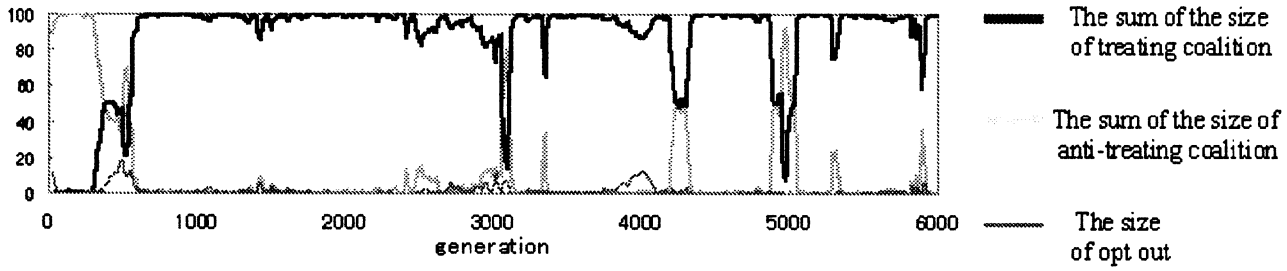


Figure 4: The size of *tc*,*anti-tc* and *opt-out* before the introduce of incentive

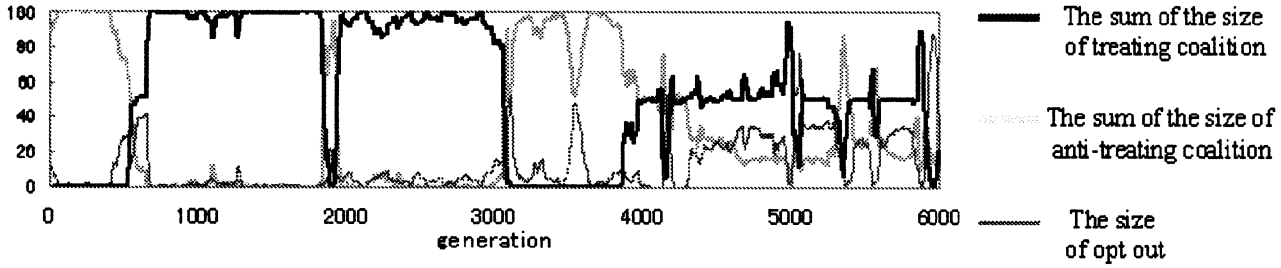


Figure 5: The size of *tc*,*anti-tc* and *opt-out* after the introduce of incentive

3 Simulation

In this paper, we introduce a subsidy to the folowing two cases to evaluate the effect of the subsidy.

The case of low payoff

In first simulation, we introduce incentive to case that treating coalition is not formed and the average payoff of factory is low. The parameters for this simulation are set as follows;

number of factories	F=100
number of lakes	$L = 3$
number of generations	$Gen = 6000$
number of iterations per generation	$I = 15$
cost for treating its wastes	$B = 10$
cost for purifying its own water supply	$D = 2$
coefficient of characteristic function	$\alpha = 600$
coefficient of characteristic function	$\beta = 5.0$
limit of the size of anti-treating coalition	$L_1 = 5$
limit of the sum of factory	$L_2 = 45$
cost of moving	$C_{move} = 20$
constant subsidy	$const_{subsidy} = 50$

Figure 2 shows the transition of the sum of the size of traeting coalition(*tc*),anti-treating coalition(*anti-tc*) in each lake and *opt-out* in case that *tc* is not formed without incentive. Figure 3 shows the transition of the sum of the size of *tc*,*anti-tc* in each lake and *opt-out* in case after the introduce of incentive.

⟨*Fig*2⟩ By 2000 generation, almost factory participated in *tc*. At 2000 generation, the size of *tc* decreased to about fifty. The size of *tc* increased at 3000 generation again. After that, almost factories selected *anti-tc* and *opt-out*.

⟨*Fig*3⟩ From 800 generation, almost factories participated in *tc*. Although, at 4200, 4900, 5200 generation, the size of *anti-tc* rapidly increased, almost factories stably continued to participate in *tc*

The case of high payoff

In second simulation, we introduce incentive to case that treating coalition is formed and the average payoff of factory is high.

Figure 4 shows the transition of the size of *tc*,*anti-tc* and *opt-out* in case that *tc* is formed without incentive. Figure 5 shows the transition of the size of *tc*,*anti-tc* and *opt-out* in case after the introduce of incentive.

⟨*Fig*4⟩ From 700 generation, almost factories participated in *tc*. Although, at 3000, 4200, 5000 generation, the size of *anti-tc* rapidly increased and almost factories continued to participate in *tc*

⟨*Fig*5⟩ From 800 to 3000 generation, almost factories participated in *tc*. From 3000 to 4000 generation, almost factories selected *anti-tc* and *opt-out*. After that, although about fifty factories participated in *tc*, this size is not stable and the size of *anti-tc* rapidly increased sometimes.

4 Discussion

As a result of simulation, the average payoff of factories is as follows.

treating coalition is not formed	without incentive	170.3
	with incentive	240.5/260.3
treating coalition is formed	without incentive	388.8
	with incentive	371.7/401.7

* (average payoff)/(average payoff + incentive)

According to the result of first simulation, when *tc* is not formed, the introduce of incentive leads to forming *tc* and increases the average payoff of factories. Contrary, according to the result of second simulation, when *tc* is formed without incentive, the introduce of incentive prevents *tc* forming and decreases the average payoff of factories. In case with incentive, because of adding subsidy, *tc* can acquire more payoff than case without incentive. At this time, *anti-tc* can acquire some payoff because *tc* is formed. Here, local governments stop giving subsidy, and then the payoff of treating coalition decreases. As a result, factories participate in *anti-tc* instead of *tc*. Incentive to form *tc* gives *tc* a subsidy, on the other hand, supports *anti-tc*. This phenomenon appears well in second simulation with incentive after 4000 generation.

5 Conclusion

We proposed the game model extending "The Lake" as agent based iterated coalition formative game. Throughout the simulations to compare the model without incentive and the model with incentive, we can confirm that incentive always does not lead to form treating coalition, sometimes lead to form anti-treating coalition.

References

[1] T.Yamashita,K.Suzuki and A.Ohuchi, " Agent based Iterated Multiple Lake Game with Local Governments", Complexity International volume5(http://www.csu.edu.au/ci/vol5/ci5.html),1998

[2] Shapley.L.S and M.Shubik, "On the Core of Economic Systems with Externalities", The American Economic ReView,vol 59,1969

[3] D.McFadzean and L.Tesfatsion, "A C++ Platform for the Evolution of Trade Networks", Working Paper,Depertment of Economics,Iwoa State University,Ames,1995

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Estimating an independent component based on the mutual information against the residual component

Takashi IWAMOTO

Advanced R&D Center
Mitsubishi Electric Corporation
Amagasaki, Hyogo 661-8661
iwataka@qua.crl.melco.co.jp

Abstract

Recently much work has been done on blind source separation (BSS) in which the sensor signals are linear mixture of independent source signals. Since optimization problems whose solution gives a separating matrix are incorporated, the number of sensor signals has been bounded from below by that of source signals. The author introduces a scheme which allows multiple solutions utilizing the known minimum value of the mutual information, and shows the experimental results of this scheme on the separation of four source signals from three sensor signals.

Key words: blind source separation, independent component analysis, optimization, mutual information, residual component

1 Introduction

Increasing attention is received by independent component analysis (ICA), for BSS, the separation of independent components from sensor signals $\mathbf{x}(t) = (x_1(t), \ldots, x_n(t))^T$ into which unknown source signals $\mathbf{s}(t) = (s_1(t), \ldots, s_n(t))^T$ have been superimposed by an unknown non-singular matrix $\mathbf{A} \in \mathbf{R^{n \times n}}$ as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t).$$

BSS is attained by finding a separating matrix $\mathbf{W} \in \mathbf{R^{n \times n}}$ which makes each component of the output

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t)$$

statistically independent with respect to each other. Though much work has been done on ICA algorithms, the common limitation that the number of sensor signals is not smaller than that of sensor signals has been imposed[5,4,2,3,1]. This is very strong condition especially for systems working in uncontrollable environment.

2 From optimization point of view

Some ICA algorithms for estimating $\mathbf{W}$ are derived from minimization of the Kullback-Leibler divergence or the mutual information

$$\begin{aligned} D(\mathbf{W}) &= \int p(\mathbf{y}) \log \frac{p(\mathbf{y})}{\prod_{i=1}^{n} p_i(y_i)} d\mathbf{y} \\ &= \sum_{i=1}^{n} H(y_i) - H(\mathbf{y}) \\ &\equiv MI(y_1, \ldots, y_n), \end{aligned}$$

where $p_i(y_i)$ is the marginal probability density function of y_i and $H(y_i)$ is the marginal entropy of y_i [4,1], since the mutual information is non-negative and becomes zero when each component y_i is statistically independent.

The author introduces a scheme to divide a sensor signal $\mathbf{x}(t)$ into an independent component $\mathbf{f}(\mathbf{x})$ separated by a separating function $\mathbf{f}$ and the residual component $\mathbf{x} - \mathbf{f}(\mathbf{x})$; we utilize the optimization problem of the mutual information $MI(\mathbf{f}(\mathbf{x}), \mathbf{x} - \mathbf{f}(\mathbf{x}))$, which directly represents the dependency between both components, over the space of the function $\mathbf{f}$. This scheme expands the optimization to allow multiple solutions.

This scheme works in tasks to separate independent components from sensor signals whose number is less than that of the source signals in principle, though the landscapes of the mutual informations over a function space may become very rugged with many minima. In practice it is impossible to carry out the optimization over the whole function space. Thus restriction on subspace is necessary. However, once we find a separating function in the subspace we can estimate independent components due to the known minimum value. In this paper let us focus on the subspace spanned by the

functions with a time delay parameter T

$$f(t;T) = \sum_i x_i(t + iT)$$

which is simple implementation of microphone array systems[6].

3 Experimental results

The scheme is tested on the generated data which emulates sampling voices of four persons with three microphones separated 40cm in a line. The voice of each persons stating his name standing 3m 50cm in front of the microphone is recorded at 48kHz into a sequence of amplified voltages $v_j(t), (j = 1, 2, 3, 4)$. Each sequence is superimposed with a fixed delay to yield signals

$$x_i(t) = \sum_{j=1}^{4} v_j(t - i\tau_j), (i = -1, 0, 1)$$

where $(\tau_1, \tau_2, \tau_3, \tau_4) = (-27, 0, 27, 54)$ emulate the angles between the line of microphones and the direction of the person from the microphones of $(60°, 90°, 120°, 180°)$ respectively. Three generated signals are plotted in Figure 2.

The entropy and the marginal entropy is calculated from the histograms of $(f(t;T), x_0(t) - f(t;T))$, $f(t;T)$, and $x_0(t) - f(t;T)$ and the obtained mutual information from them is plotted against the time delay parameter, in Figure 1. The component $f(t, 1)$, which corresponds to the second valley from the left in the landscape shown in Figure 1, is plotted in Figure 3. The source signal is plotted in Figure 4.

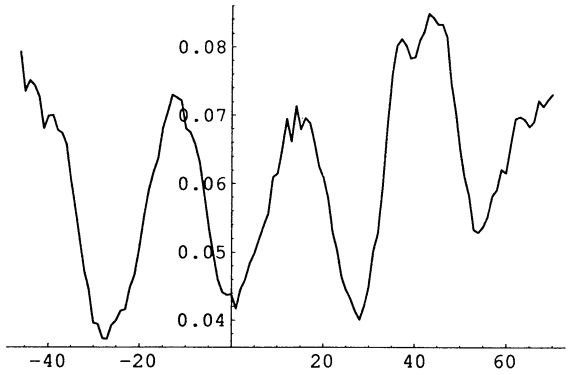


Figure 1: The landscape of the mutual Information against the time delay.

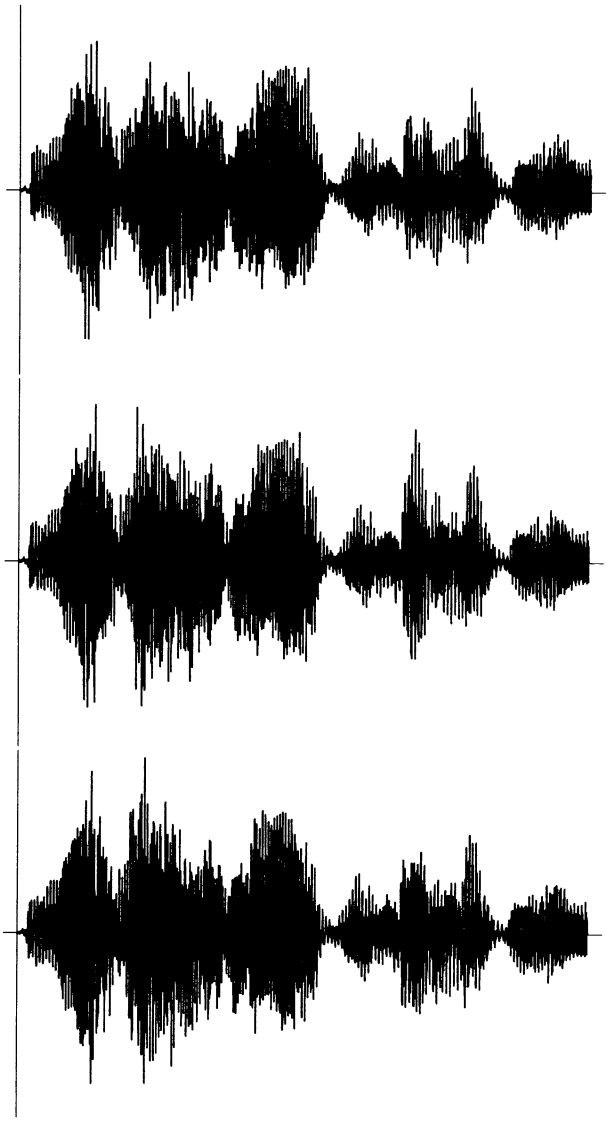


Figure 2: Three sensor signals.

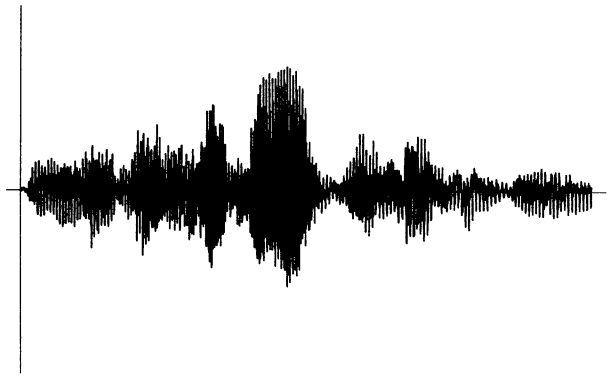


Figure 3: Extracted signal.

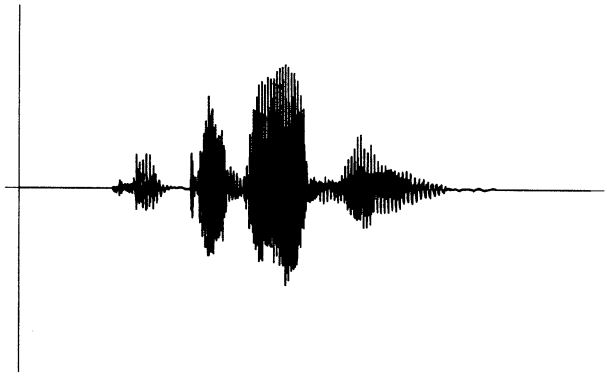


Figure 4: Source signal (voice pronouncing Ta-Ke-Da-Jun).

4 Conclusion

The author introduces a scheme which allows multiple solutions utilizing the known minimum value of the mutual information. Experimental results shows that this scheme separetes four source signals from three sensor signals.

Acknowledgements

The author is grateful to Mr. Ikeda and Mr. Takeda for data sampling and furuitfull discussions.

References

[1] Amari S, Cichocki A, and Yang HH(1996), A new learning algorithm for blind signal separation, In: Advances in Neural Information Processing Systems 8, MIT Press, pp.757-763

[2] Bell AJ and Sejnowski TJ(1995), An information-maximization approach to blind separation and blind deconvolution, Neural Computation 7:1129-1159

[3] Cardoso J-F and Laheld BH(1996), Equivariant Adaptive Source Separation, IEEE Trans. on Signal Processing, 44:3017-3030

[4] Comon P(1994), Independent component analysis, a new concept? Signal Processing, 36:287-314

[5] Jutten J and Herault J(1991), Blind separation of sources, PartI: An adaptive algorithm based on neuromimetic architecture, Signal Processing, 24:1-10

[6] Pillai SU(1989), Array signal processing, Springer-Verlag

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Rotational Model of the Stock Prices (I)

Satoshi Maekawa* Yoshi Fujiwara†
Auditory and Visual Informatics Section
Communications Research Laboratory
Iwaoka 588-2, Nishi-ku, Kobe 651-2401, Japan

Abstract

We observe joint probability distribution of successive logarithmic returns of the daily stock prices, and found that it has asymmetric form which is a rotation of independent distribution. This means that successive price changes are statistically dependent. We apply independent component analysis (ICA) to decompose the stock price changes into independent components, and obtained the impulse response for each component.

Because the impulse responses are almost olthogonal, we try to model it with a rotation filter. We show that the rotational model can explain a long-term fine structure of the impulse response for decades of days. This impulse response shows that the price change always has overshooting. We suggest that this is an observation of rational bubbles mentioned by Mandelbrot.

1 Introduction

In Fama's semi-strong form of efficient market hypothesis(EMH)[1], public information on an asset is assumed to be channelled immediately into its price.

However, this hypothesis is doubtful because all participants cannot obtain all information always in time and it can be imagined the time will differ according to an individual participant, from putting information in hand to the reflection of it in the actual buying and selling. So it is natural to assume that each information will continue to influence on the market for a while.

If the market is formed under such condition, the impulse response for each informaion is not orthogonal, so the time-series is not martingale. Such market does not satisfy even the weak form of the EMH, in which all market information is reflected in prices. This problem is avoided by introducing rational bubbles which are pointed out by Mandelbrot[2] and Blanchard et al.[3]. Then, the rational bubbles must occure if the information influences have different time-lags and the market satisfy the weak form of EMH, that is, there are some arbitragers.

In this paper, we attempt to evaluate the influence to the market with such time-lags quantitatively by using independent component analysis (ICA)[4][5]. In using ICA for the analysis of the stock prices, we assume that 1) information influencing to the market is independent, 2) the probability distribution of the magnitude of influences on the market is leptokurtic, and 3) the infuluences of some informations are linearly accumulated. The first assumption is trivial by definition of information, and the second and third are not obvious but can be thought to hold empirically.

In chapter 2, a statistical character of stock prices change is described and its probability density is observed. Chapter 3 explains ICA, and shows the extracted impulse respnse. In chapter 4, this impulse response is modeled by using a rotation filter, and the satisfactory coincidence with the actual impulse response is obtained. Chapter 5 describes conclusion.

2 Statistics of the stock prices

Our study is based on the analysis of actual time-series data of daily stock prices consisting of 387 companies from September 1987 to April 1998 (Tokyo Stock Exchange). First of all, we observe a statistical character of logarithmic returns. Logarithmic return r_t is given by

$$r_t = \log(\frac{x_t}{x_{t-1}}), \quad (1)$$

where x_t is a stock price at time t. The mean value of the logarithmic returns of all companies is -2.90×10^{-4} and the standard deviation is 2.52×10^{-2}. The probability distribution function of the logarithmic returns is not a normal distribution, and the kurtosis is 11.2.

*E-mail: maekawa@crl.go.jp
†Domestic Research Fellow. E-mail: yfujiwar@crl.go.jp

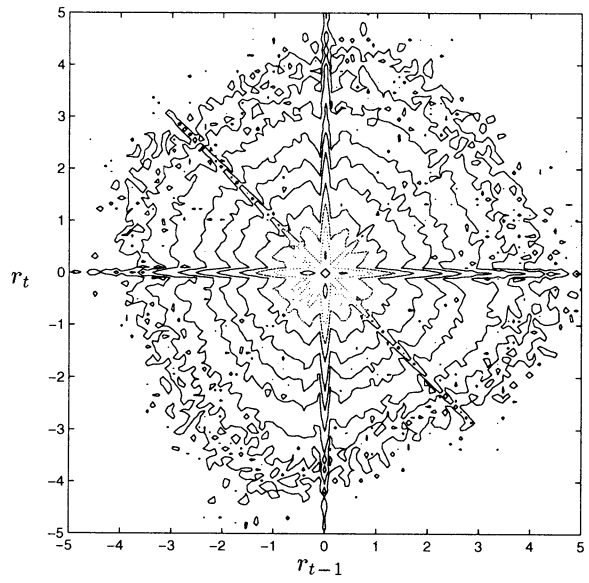


Figure 1: Joint probability distribution of the successive returns. The variance is normalized to 1 for each return.

The joint probability distribution of successive returns is shown in Fig.1. The scale is normalized by the standard deviation. The high probability density part can be observed on each coordinate axis and in each direction of 45 degrees, because the stock value is quantized. We would be able to ignore this effect.

Asymmetry is found in this figure which is like rotating independent distribution clockwise by about ten degrees. This asymmetry suggests that the stock price changes are statistically dependent.

3 Independent Component Analysis

Recently, there is an active research of the technique of ICA which separates a set of signals into statistically independent components by using higher-order statistics. Here, we apply this method to financial time-series.

The n-dimensional windowed signal, $\{r_{t-n+1}, \cdots, r_{t-1}, r_t\}$ is to be an input, and ICA is applied to it,

$$y_{t,i} = \sum_{j=1}^{n} w_{ij} r_{t-j+1}, \quad 1 \leq i \leq n. \tag{2}$$

Here, $y_{t,i}$ is independent component and w_{ij} is demixing matrix. Let $(W)_{ij} = w_{ij}$, $\boldsymbol{y}_t = (y_{t,1}, \cdots, y_{t,n})^T$,

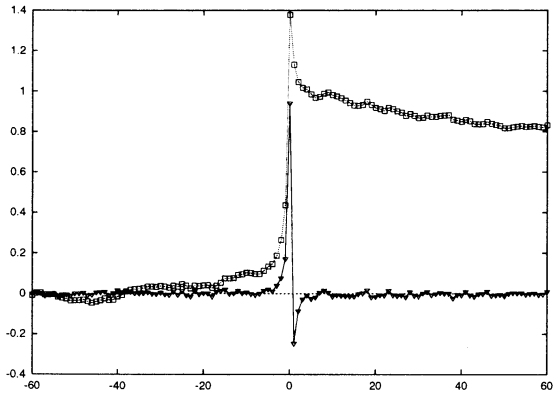


Figure 2: The impulse response (solid line) and its cumultive curves (dotted line).

$\boldsymbol{r}_t = (r_t, \cdots, r_{t-n+1})^T$, then Eq.(2) is written as:

$$\boldsymbol{y}_t = W \boldsymbol{r}_t. \tag{3}$$

The ICA requires that $y_{t,i}, (i = 1, \cdots, n)$ is independent and its probability distribution $p(y_{t,i})$ is leptokurtic (or platykurtic). In the case of stock market, it might be assumed that the information influencing to the market is independent and the probability distribution of the magnitude of influences on the market is leptokurtic.

We adopt the learning rule of ICA proposed by Bell and Sejnowski[4] and natural gradient method by Amari[5]. The windowed signal $\boldsymbol{r}_t$ are randomly sampled from all stocks and period, and W is updated for every sample as follows:

$$\Delta W = \varepsilon \left(I - f(\boldsymbol{y}) \boldsymbol{y}^T \right) W, \tag{4}$$

where ε is a coefficient of learning, and I is a unit matrix. The notation $f(\boldsymbol{y})$ is defined by

$$f(\boldsymbol{y}) = (f(y_1), f(y_2), \cdots, f(y_n))^T. \tag{5}$$

The function $f(\cdot)$ is given by:

$$f(y) = \frac{2}{1 + \exp(-y)} - 1. \tag{6}$$

3.1 Results of ICA

The result of ICA is described. Fig.2 shows the time-series of the impulse response extracted by ICA. This plot corresponds to a row vector of W^{-1}. Here the plot is averaged over all rows under the condition that the peak is aligned, because each row vector has symmetry along time-axis.

This plot shows that the price change always has overshooting, as if traders seem to overreact to each

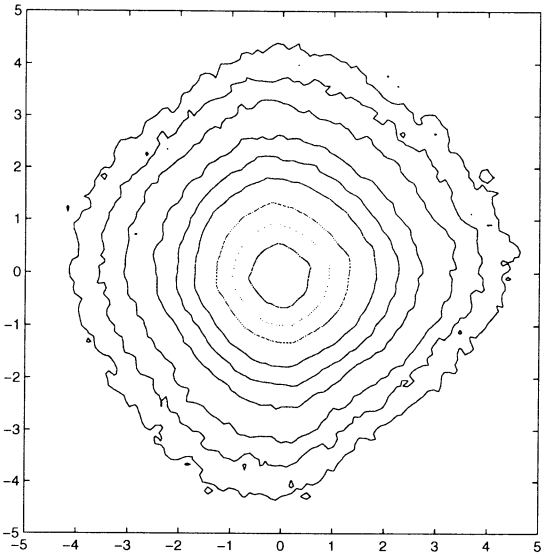


Figure 3: Joint probability distribution of two successive independent components.

information. We also observe autocorrelation of the impulse response and it shows that this signal is highly orthogonal. The orthogonality is needed for market to be efficient. These suggests that this overshooting behaviour corresponds to the so-called rational bubble.

Fig.3 shows the joint probability distribution of two successive independent components. It is understood that each swelling of the probability distribution lies in the direction of coordinates axis. This distribution can be seen to be more independent than the original distribution(Fig.1).

4 Rotational model of the stock prices

In this section, a model is proposed which describes the impulse response extracted by ICA which are highly orthogonal. Fig.1 suggests that the rotated joint distribution becomes independent.

4.1 Rotational model

A filter which uses the following rotations is composed. In the rotation filter, the impulse response is completely orthogonal. First of all, $n \times n$ rotation matrix $R_j(\theta)$ is defined as follows:

$$R_j(\theta) = \begin{array}{c} \\ 1 \\ \\ \\ j+1 \\ \\ \\ n \end{array} \begin{array}{c} \begin{array}{cccccccc} 1 & & \cdots & & j+1 & & \cdots & n \end{array} \\ \begin{bmatrix} \cos\theta & 0 & \cdots & 0 & -\sin\theta & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & & & & & \\ 0 & 0 & \cdots & 1 & 0 & 0 & \cdots & 0 \\ \sin\theta & 0 & \cdots & 0 & \cos\theta & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & & & & \ddots & \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \end{array}, \tag{7}$$

and n-dimentional matrix $\tilde{R}(\boldsymbol{\theta})$ with parameters $\boldsymbol{\theta} = (\theta_1, \cdots, \theta_{n-1})^T$ is multiplied successively:

$$\tilde{R}(\boldsymbol{\theta}) = \prod_{j=1}^{n-1} R_j(\theta_j). \tag{8}$$

Then a $N \times N$ rotation matrix $R_t(\boldsymbol{\theta})$ is composed as follows:

$$R_t(\boldsymbol{\theta}) = \begin{bmatrix} I_{N-t-n+1} & 0 & 0 \\ 0 & \tilde{R}(\boldsymbol{\theta}) & 0 \\ 0 & 0 & I_{t-1} \end{bmatrix}. \tag{9}$$

Here, I_m is an $m \times m$ unit matrix. By repeatedly multiplying this rotation matrix, we get

$$R(\boldsymbol{\theta}) = \prod_{t=1}^{N-n+1} R_t(\boldsymbol{\theta}). \tag{10}$$

Each row vector of of $R(\boldsymbol{\theta})$ becomes a filter to statistically independent noise, $\hat{\boldsymbol{y}}$.

$$\hat{\boldsymbol{r}} = R(\boldsymbol{\theta})\, \hat{\boldsymbol{y}} \tag{11}$$

4.2 Analysis of rotational model

The relation between the rotation filter and the observed impulse response is considered. First of all, all parameters of rotation filter, $\{\theta_i\}$, are completely determined by fitting with the $t < 0$ part of the observed impulse response. With these parameters, we observe the rest part of the impulse response generated by the filter. Fig.4 shows the $t > 0$ part of the impulse responses generated by the rotation filter and that observed by ICA. Fig.5 shows the results of correlation analysis of both signals. These figures show that they are suprisingly matching with each other for decades of days. This matching suggests that 1) the impulse response is highly orthogonal, and 2) the small value of the impulse response might not be a mere noise and meaningful even far from the peak. The former is obvious from autocorrelation, but the later is an amazing fact if it is true. Further study is needed.

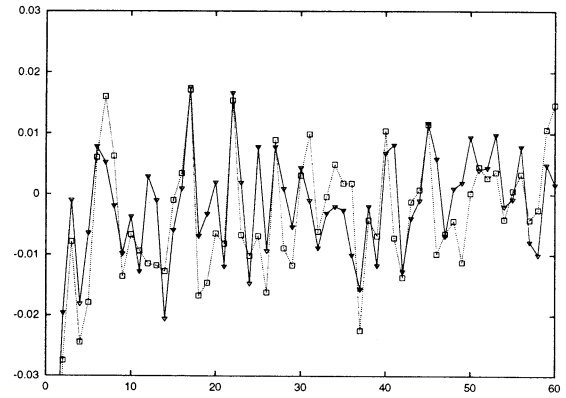


Figure 4: The $t > 0$ part of of the impulse responses. The solid line shows the impulse response generated by the rotational model and the dotted line shows the observed one with ICA.

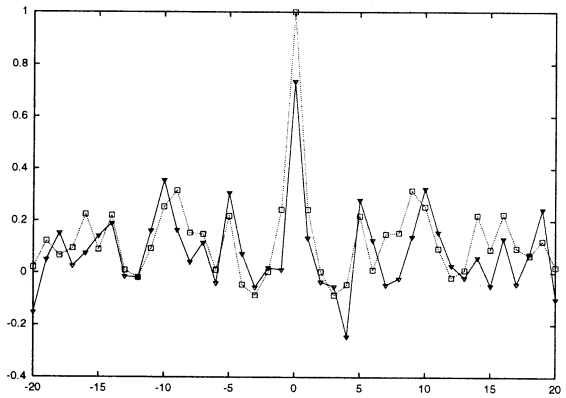


Figure 5: The solid line shows cross-correlation of impulse responses with ICA and rotational model. The dotted line shows auto-correlation of the observed one with ICA. Both are analysed on period, $4 \leq t \leq 60$.

5 Summary

In this paper, we observed the joint probability distribution of successive logarithmic returns and found that they are not independent. Moreover, impulse response is extracted by ICA as an indepent component, and we showed that it can be approximated very well with a rotational model for decades of days. This result supports the weak form of the EMH.

The overshooting property of the price changes led from the efficiency of the market is known as rational bubbles[2][3]. So, it can be said that we could observe the long-term fine structure of the rational bubbles in the real market data by using ICA. However, since the observed bubbles continue for only several days, all bubbles are not covered in the analysis.

Some mechanisms are proposed to explain the cause of rational bubbles, but in the case analyzed here the non-uniformity of information spread and the existance of arbitragers can be considered.

We hope that these quantitative observation will be useful to understand the market mechanism as complex society.

Acknowledgements

Yoshi Fujiwara is supported by Domestic Research Fellowship of Japan Science and Technology Corporation.

References

[1] E. F. Fama., "Efficient capital markets: A review of theory and empirical work," *Journal of Finance*, pp. 383–417, 1970.

[2] B. B. Mandelbrot, "Forecasts of future prices, unbiased markets, and 'martingale' models," *Journal of Business*, Vol. 39, 242–255, 1966.

[3] O. J. Blanchard and M. W. Watson, "Bubbles, rational expectations, and financial markets," In *Crises in the Economic and Financial Structure*, P. Wachtel(ed.), Lexington Books, 1982.

[4] A. J. Bell and T. J. Sejnowski, "An information maximisation approach to blind separation and blind deconvolution," *Neural Computation*, Vol. 7, pp. 1129–1159, 1995.

[5] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," In *Advances in NIPS*, Vol. 8, MIT Press, pp.757–763, 1996.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Rotational Model of the Stock Prices (II)

Yoshi Fujiwara* Satoshi Maekawa†
Auditory and Visual Informatics Section, Communications Research Laboratory,
Iwaoka 588-2, Nishi, Kobe 651-2401, Japan

Abstract

Stock market is a complex system comprising a large number of human activities. We attempt to solve an *inverse problem* — to infer herd behavior and information propagation in market, specifically, rational bubble and its heterogeneity. We argue that the significant pattern in stock prices discovered in the accompanying paper is related to how the impact of information is reflected in a long-time behavior of prices. We show a few statistical quantities, how they are related to the rotational model we proposed, and its implication in the structure of rational bubble in an efficient market.

1 Introduction

Understanding and possible control of complex systems are inseparable goals of science and technology. Financial market is one of the large-scale human activities that has been challenging researchers and practitioners. This paper, through the study of actual stock prices in market, attempts to uncover a new property of this complex system.

Market agent or participant takes action on the basis of expectations in attempt to maximize returns. *Expectation* loosely means one's prediction on future position that one may find oneself in. In a stock market an important factor determining one's expectation is fundamental values of future and current dividends and interest rates. If many participants' expectations are based solely on these *fundamentals*, the asset price will be tranquilized to an equilibrium point. However, this picture would not be borne out in a liquid (highly active) market. Rather market participant often believes that many others speculate a directional deviation of an asset price from that determined by fundamentals. As an example, suppose that many people believed a rise of price, not necessarily based on fundamentals, ran to buy *and* the rise is realized. Then the participants whose prior expectation coincides with the realization can obtain a return of capital gain, which is a good motivation for the initial speculation. This so-called *self-fulfilling speculation* is an important factor of participant's expectation and behavior in a liquid market.

Rational expectation in assessing an asset price at a certain time includes both the current and future evaluation of fundamentals and expectation of capital gain. Deviation from from a price determined by fundamentals due to rational expectation and self-fulfilling speculation is called *rational bubbles** [1][2]. It would be possible that rational bubbles can take place even in a *heterogeneous* market. That is, market participant's expectation and behavior is determined by a limited quantity of information of each own. They will interpret news and react to it in different time-scales and frequencies.

We attempt to find *statistical properties* universally observed in financial data, which give some useful information on the mechanism and dynamics of the agents involved. In the accompanying paper [13] (hereafter referred to as **I**), we discovered a significant pattern in the temporal structure of rates of returns of stock prices by using a recently developed technique in signal processing, independent component analysis (ICA). We will argue that it is the trace of rational bubbles that occur frequently in daily price changes. Additional but crucial observation is that efficient market hypothesis in a weak sense holds. These observations yield insight into the self-fulfilling speculation and rational bubble in an efficient market.

In Section 2 we show a few statistical quantities observed in the data. In Section 3 we argue how they are related to the rotational model we proposed and its implication in the structure of rational bubble and its heterogeneity. Conclusion and discussion is given in Section 4. We use the same notation as in **I** throughout this paper.

*Domestic Research Fellow. e-mail: yfujiwar@crl.go.jp
†e-mail: maekawa@crl.go.jp

*This is different from irrational bubbles caused by enthusiasm or panic, which are completely free from fundamentals.

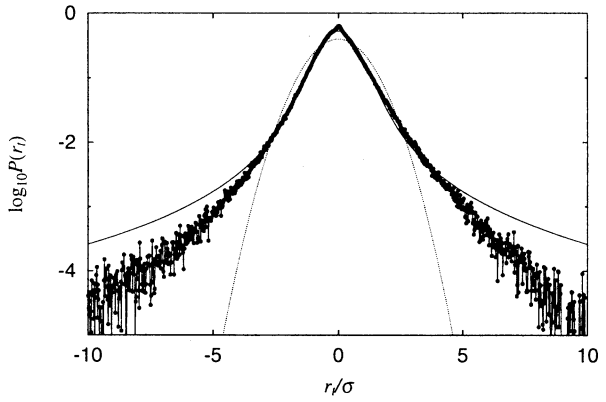


Figure 1: PDF for logarithmic returns. Points represent the distribution obtained after ICA. Dotted line is a Gaussian distribution with the same variance as the sample variance σ^2 of the data. Thin line is a stable distribution (symmetric).

2 Statistical Properties

A few statistical properties in the data of **I** are:
(1) *pdf of* r_t
The probability density function (pdf) of logarithmic returns after ICA has a non-Gaussian character. Figure 1 shows the pdf and a Gaussian distribution with the same sample variance. Also shown is a trial fit of a symmetric stable Pareto-Lévy distribution. Stable distribution is defined by a property of self-similar invariance under summation†.

Stable distribution for returns was first discovered by Mandelbrot in [3][1] and later observations confirm and modify the picture of the heavy-tail behavior (see [4] for example). The pdf is well fitted by the stable distribution, but with a more rapid decay in the heavy tail ($\geq 4\sigma$). (Note that recent observation is done with high-frequencies from seconds to hours while ours is daily.) There are proposed many models including truncated Lévy flight [4], multiplicative process [5], stretched exponential distributions [6].
(2) *Autocorrelation*
The sample autocorrelation function for r_t is shown as a bold line in Figure 2(a). (This is the average over all the stock names described in **I**.) Although it

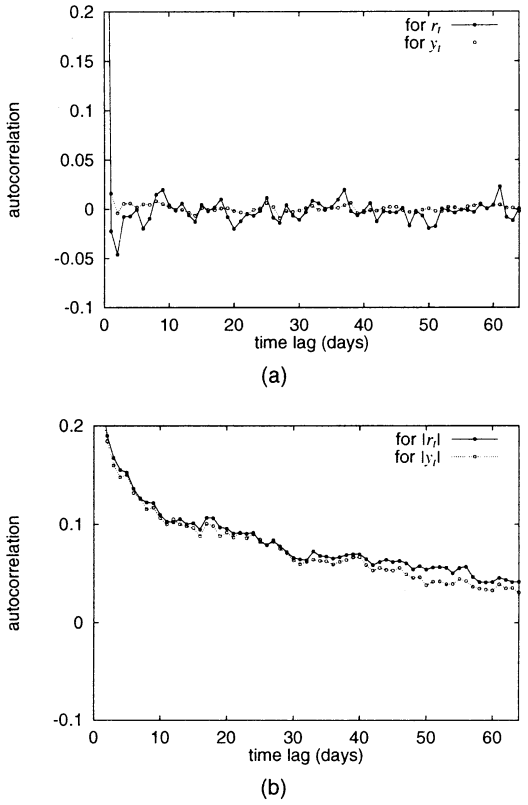


Figure 2: Autocorrelation functions for (a) r_t and (b) $|r_t|$. Time lag is given in days. In each figures, filled circles are for raw data (averaged over all the stock names in the dataset) and empty circles are for filtered data by ICA.

is not generally easy to test the statistical significance of autocorrelation, it would be safe to say that there is no significant correlation except the first few lags (1,2 days). This is necessary for efficient market hypothesis in the sense of "martingale"‡to hold.

However, no correlation, or equivalently, spectral whiteness, is not synonymous with statistical independence in the case of non-Gaussian. Statistical information of phase is completely lost in these statistics. Indeed, ICA in **I** extracted the *phase information* as we shall explain below.

†Its pdf has four parameters. Ignoring the two representing translation and skewness, its characteristic function is

$$\phi(t) = \exp(-\gamma^\alpha |t|^\alpha), \qquad 0 < \alpha \leq 2,\ 0 < \gamma.$$

The fit in Figure 1 is given by $\alpha = 1.6$ and $\gamma = 0.54$. Skewness parameter should be included but is not further examined here.

‡A martingale process for a sequence of stochastic variables $\{x_t\}$ $(t = 1, 2, \cdots)$ is defined by the condition that $E\{|x_t|\}$ is finite and that $E\{x_t|x_{t-1}, \cdots, x_1\} = x_{t-1}$, where $E\{\cdot|\cdot\}$ is the conditional expectation. Intuitively, this means "you can't do better than predicting so that tomorrow's price is the same as today's". It is not an easy task to verify the martingale property from a given data.

(3) *Volatility clustering*
The statistical dependence includes the so-called clustering of volatility (variance). Our concern in this work is not relevant to this phenomenon as we shall argue, but is worth mentioning here. If we were to have statistical independence, we should expect that a sample autocorrelation function for any nonlinear functions of r_t should decay exponentially as time lag increases. Figure 2(b) plots the autocorrelation for the absolute value of r_t. It is obvious that the decay is surprisingly slow, and is actually hyperbolic. See [7] and references therein.

3 Phase Information and Rotational Model

Let us take a look at the rotational model in **I** and how it is related to the above statistical properties.

First of all, the absence of correlation observed in the preceding section strongly implies the presence of *arbitragers*, who promptly eliminate any advantages to buy and sell based on future extrapolation. They could do so by judging significance of tiny serial correlation and by exploiting it. Several scenarios of arbitraging processes that makes a market efficient have been considered in [8][9][1][2], for example. On the other hand, participants with rational expectation attempt to do self-fulfilling speculation and behave in various time-scales and frequencies. Basically, only when information that cannot be extrapolated from current and past expectation and fundamentals is given to the market, it will be a significant cause to price changes reflecting the behaviors of both arbitragers and speculators.

Now from successive n data with time lag of 1 (day), construct a "lag vector" $\boldsymbol{r}_t = (r_t, r_{t-1}, \cdots, r_{t-(n-1)})$. We can regard the whole data $\{r_t\}$ as a bunch of points $\{\boldsymbol{r}_t\}$ in n dimensional space. A pdf for the distribution of these points can be defined and denoted as $p_{\boldsymbol{r}}(\boldsymbol{r}_t)$. Its projection on 2-dimension was Figure 1 in **I**.

The above picture naturally leads us to consider a decomposition

$$r_t = \sum_{s=-\infty}^{t} \lambda_{t-s}\nu_t$$

in terms of cause, or innovation ν_s and its lagged response λ_t. As an a priori assumption, we suppose that $\{\nu_s\}$ are independent, but not necessarily Gaussian. Let us then try to find an inverse filtering

$$y_u = \sum_{t=1}^{n} W_{ut}\, r_t, \text{ or } \boldsymbol{y}_u = W\,\boldsymbol{r}_t, \qquad (1)$$

so as to decompose the price change into *independent* innovations and their impulse response. The problem is to infer W such that the pdf $p_{\boldsymbol{y}}(\boldsymbol{y}_u)$ transformed from $p_{\boldsymbol{r}}(\boldsymbol{r}_t)$ as

$$p_{\boldsymbol{r}}(\boldsymbol{r}_t; W) = |\det W|\, p_{\boldsymbol{y}}(\boldsymbol{y}_u) \qquad (2)$$

becomes close to the independent combination, $\prod_{i=1}^{n} p_y(y_u)$. We have little knowledge about $p_y(y_u)$ except that the distribution of such innovations would be leptokurtic. This is a semi-parametric estimation problem, whose theoretical foundation is only recently established. ICA is an implementation of it.

Let us remark that situation is a little complicated due to the presence of volatility clustering. That is, although each information comes into a market independently (by its definition), successive components in the actually reflected price change are not completely independent. Indeed, volatility clustering related to Figure 2(b) gives a significant structure of long time-scale, which needs further study. However, the dotted line in Figure 2(b) implies that ICA does not affect the structure much. So we think the impulse response in **I** gives insight of rational bubbles, little affected by the volatility clustering.

The resulting filter in the form of impulse response W^{-1} was given in Figure 2 of **I**. Since the same ensemble is seen in the two frames, $\{r_t\}$ and $\{y_t\}$, we have

$$\begin{aligned} \langle\, r_t r_s \,\rangle &= \sum_{u,v} (W^{-1})_{tu}(W^{-1})_{sv} \langle\, y_u y_v \,\rangle \\ &= \sum_{u \neq v} (W^{-1})_{tu}(W^{-1})_{sv} \langle\, y_u y_v \,\rangle \\ &\quad + \langle\, y^2 \,\rangle \sum_{u} (W^{-1})_{tu}(W^{-1})_{su} \end{aligned} \qquad (3)$$

where $\langle\, \cdot \,\rangle$ denotes the ensemble average. In the second line we used the temporal homogeneity. Since ICA makes $\{y_t\}$ orthogonal, the first term vanishes. (See the dotted line in Figure 2(a).) Thus the absence of correlation for $t \neq s$ means that the impulse response W^{-1} is an orthogonal matrix. Due to the temporal homogeneity, W_{ut} should be a function of the lag $u-t$ only. Therefore, W is a rotation with only n parameters. This was the rotational model in **I**.

On the other hand, non-vanishing filter coefficients in W^{-1} represent the phase information that is not contained in autocorrelation and spectral analysis — how the arbitraging process and accompanying heterogeneous market activity takes place. Though it is beyond the present manuscript, we comment that this process may be modeled by a multiplicative stochastic

process, the simplest form of which was given by [2]. It might be related to the pdf of returns since the process with a nonlinear effect generates a stable distribution with rapid decay in heavy tails [4][5].

4 Conclusion and Discussion

When one goes beyond equilibrium theory based on fundamentals, one has to consider an infinite number of possibilities for agents' expectations and behaviors. How can one choose an appropriate one? This was the serious problem addressed explicitly by economists and scientists in the Santa Fe group's computational approach of artificial market (see more in [10]).

Our approach attempts to infer it from statistical properties and patterns found in the actual data of stock market. We observed the leptokurtic pdf, vanishing autocorrelation and volatility clustering as other researchers have found. While the absence of autocorrelation strongly implies the presence of arbitragers, phase information is included in the significant pattern found in the data by ICA in **I** is the phase information which gives us insight of behaviors of speculators in the presence of arbitragers who makes the market efficient. We argue that the pattern is caused by rational bubbles, what was first pointed out by Mandelbrot, but need further study of the details.

This approach may be viewed to be complement to the computational approach in the sense that if essential mechanism and dynamics in the actual stock market are captured in those multi-agents systems, then both share the same characteristics in the form of statistical properties and ubiquitous patterns in observables. It would be very interesting to examine such features in many artificial systems such as [10][11][12] in order to understand this complex system.

Acknowledgements

We would like to thank T. Akinaga (Keio Univ.), T. Takashina (Nikon), K. Takaoka (Hitotsubashi Univ.) and H. Takayasu (Sony, CSL) for information and discussions. Numerical calculation of stable distribution was done with the program by J. P. Nolan (American Univ.). Y. F. is supported by Domestic Research Fellowship of Japan Science and Technology Corporation.

References

[1] B. Mandelbrot, *Fractals and Scaling in Finance*, Springer, 1997.

[2] O. J. Blanchard and M. W. Watson, "Bubbles, rational expectations, and financial markets", ed. P. Wachtel, *Crises in the Economic and Financial Structure*, Lexington Books, 1982.

[3] B. Mandelbrot, "The variation of certain speculative prices", *Journal of Business*, Vol. 36, pp.394-419, 1963.

[4] R. N. Mantegna and H. E. Stanley, "Scaling behavior in the dynamics of an economic index", *Nature*, Vol. 376, pp.46-49, 1995.

[5] H. Takayasu, A.-H. Sato and M. Takayasu, "Stable infinite variance fluctuations in randomly amplified Langevin systems", *Physical Review Letters*, Vol. 79, pp.966-969, 1997;

[6] J. Laherrere and D. Sornette, "Stretched exponential distributions in nature and economy", *European Physical Journal* **B**, Vol. 2, pp.525-539, 1998.

[7] R. Cont, M. Potters and J.-P. Bouchaud, "Scaling in stock market data: stable laws and beyond", in *Scale Invariance and Beyond*, Proceedings of the CNRS Workshop on Scale Invariance, Les Houches, 1997; R. Cont, "Scaling and correlation in financial time series", AFFI International Finance Conference, Grenoble, 1997.

[8] B. Mandelbrot, "Forecasts of future prices, unbiased markets, and "martingale" models", *Journal of Business*, Vol. 39, pp.242-255, 1966.

[9] B. Mandelbrot, "When can price be arbitraged efficiently? A limit to the validity of the random walk and martingale model", *Review of Economics and Statistics*, Vol. 53, pp.225-236, 1971.

[10] W. B. Arthur, J. H. Holland, B. LeBaron, R. Palmer and P. Tayler, "Asset pricing under endogenous expectations in an artificial stock market", SF working paper 96-12-093; eds. P. W. Anderson et al., *The Economy as an Evolving Complex System*, Addison Wesley, 1988.

[11] P. Bak, M. Paczuski and M. Shubik, "Price variations in a stock market with many agents", *Physica* **A**, Vol. 246, pp.430-453, 1997.

[12] A.-H. Sato and H. Takayasu, "Dynamic numerical models of stock market price", *Physica* **A**, Vol. 250, pp.231-252, 1998.

[13] S. Maekawa and Y. Fujiwara, "Rotational Model of the Stock Prices (I)", *in this volume.*

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Multiagent Approach against Computer Virus: An Immunity-Based System

Takeshi Okamoto
Nara Institute of Science and Technology
Yoshiteru Ishida
Toyohashi University of Technology, Tempaku, Toyohashi, 441-8580 Japan,
ishida@tutkie.tut.ac.jp

Abstract

This paper proposes a biologically inspired detection and repair system against computer viruses. Similarly to the biological protection systems, our system is composed of a subsystem corresponding to the immune system and that to the repair system by copying. The former subsystem system recognizes non-self using the information of the self. It has agents similar to antibody, killer T-cell and helper T-cell. The repair system ask other computers on LAN to send a complete copy, and a propagation agent which recieves or send a complete copy.

1 Introduction

The explosive expansion of the Internet means that computer viruses obtained their infection route to the enormous amount of computers connected to the Internet. Indeed, computer viruses expanded rapidly since 1987. It is said that the number of computer viruese exceeds ten thousands, and another six viruses appeared everyday on average. To make matters worse, this trend will accelerate due to the development of the Internet [1, 2].

The origin of the concept of computer virus may be traced back Von Neumann's works on *Self-reproducing automata* in 1940. The word "computer virus" may be first found in [3] in 1984. We could not identify the first computer virus, however, it is said that the one found in a university in 1986.

After the debut of computer virus, many anti-virus programs have been developed. However, most of them uses information specific to virus for detection and repair, they cannot treat unknown new viruses. Some anti-virus program, indeed, find unknown viruses by a heuristic method. Unfortunately, it cannot detect all of them and it may even make false alarm. Further, it cannot repair the infected PC most of times. Recently, Kephart proposed a digital immune system that learned some aspects of the immune system [1]. It is a centralized approach where one computer analyze the virus and distribute the vaccination program. Consequently, once the computer is attacked or the path to/from the computer is shutdown, "the immune system" will completely break down. Our approach is a distributed approach by multiagents, and may be closer to the strategy of "the immune sytstem" in the sense that many heterogeneous agents cooperate from detection, neutralization to repair and adaptation to the second encounter. Forest devised a new detection method that learned from the self/nonself discrimination mechanism of the immune system [4]. The method potentially can find new virus, since it may recognize all the codes that do not match "the self" as "nonself". However, how to keep track of the information of the self, when it is often updated, is another problem. Pu proposed a diversification method of the implementation of OS to escape from the virus attack [5]. This diversification strategy seems to be truly biological. The strategy is indeed taken not only by the immune system but also by the sex system.

We have been working on autonomous decentralized systems based on the immune system and proposed an anormal detection and diagnosis method [6, 7, 8]. This paper proposes yet another method learned from the immune system. We use a distributed approach by heterogeneous agents to fully use network. From the first phase of virus detection to the final phase of the system repair, autonomous agents work cooperate, playing a different roles. The basic idea is quite simple;

- Since computer viruses use the network as their infection route, why not use the network for anti-virus system.
- Since computer viruses share many features from biological virus, why not use the strat-

egy by the biological defense system of multicellularorganism such as those by the immune system; i.e. cooperative works by autonomous and distributed agents.

2 Computer Virus and Anti-Virus Program: State of The Art

2.1 Computer virus

Computer virus may be defined to be a program that can infect by altering other executable programs [3]. Although there are few works to utilize computer viruses [10] or to regard them as a type of artificial life [9], they are generally harmful programs. Computer viruses may be classified in many ways based on types of hosts, symptons, infection parts, infection method, whether or not it stay in the memory after termination (TSR), and so on. Infectin parts can be; master boot record (MBR at the top of disk) and usual part of disks. Virus that can infect the usual part of disks can be further divided into the file infection virus that infect executable files and the macro virus that infect data file with macro.

Based on infection method, computer viruses can be classified to the following three types: 1) Post infection type that append the virus program to the tail of the host program; 2) Pre infection type that add the virus program at the head of the host program; and 3) Overwriting infectin type that overwrite the host program.

Although the virus body is inserted into different parts of the host program, the virus must alter at least five bytes of the head of host program to seize the execution priority for all these three types. This feature will be used in our anti-virus system (see section 3.1).

2.2 Anti-Virus program

Anti-virus programs or vaccination programs have been developed against computer viruses. The importance of these programs increases, since the number of PCs connected to LAN more rapidly increases (compared with workstations whose OS is mostly unix). Anti-virus programs use the following three types algorithms: 1) Scanner method: it scans MBR, disks, and memory to find the "signature" of virus; 2) Checksum method: it uses file checksum or similar ones to detect file alteration; and 3) Monitoring method: it monitors program execution to prevent instruction sequeces particular to computer viruses. Most anti-virus programs adopt the combination of 1) and 3). Some new version of 1) automatically generate a "prescription" by analyzing the signature [1]. Our system is based on 2), however our system also take care of repair (see section 3.2), although the method 2) itself do not handle repairing.

3 The Proposed Method

The main feature of the proposed method is that it uses a distributed approach by autonomous and heterogeneous agents. Therefore, the task is done in parallel in time and in a distributed manner in anyplace in the network. However, similarly to the immune system (we do not persue the parallelism between the immune system and our computer immune system, some agents are named after those of the immune system, though), there are qualiataively different stages; detection/neutralization stage and recovery stage.

3.1 Detection/Neutralization stage

The important feature of our system in the detection/neutralization stage is that it uses the information of "the self" (as is done by the immune system) rather than the information of "nonself" (as by the most vaccination programs). As pointed out in section 2.1, most virus programs must alter the first few bytes of the host program. Our system keeps the information of first few bytes (the parameter including how many bytes are kept can be changed adaptively) of the executable system files (that are potentially under attack) with other attributes such as file size, checksum, etc. This first few bytes with other file attributes are kept as "the self" information. The self informations are encrypted (and compressed if necessary) into a distributed database.

The antibody agents first detect file alteration by matching "the self" information with the current state of files. As the user defines the scope of "the self", there will be no false alarm as is done in the checksum method [11]. Next, the antibody agents neutralize the effect of virus by overwriting the first few bytes of the self information (Fig. 1). Although this overwriting takes back the execution priority, it is not complete recovery because the host program may be still different in the parts other than the overwritten head.

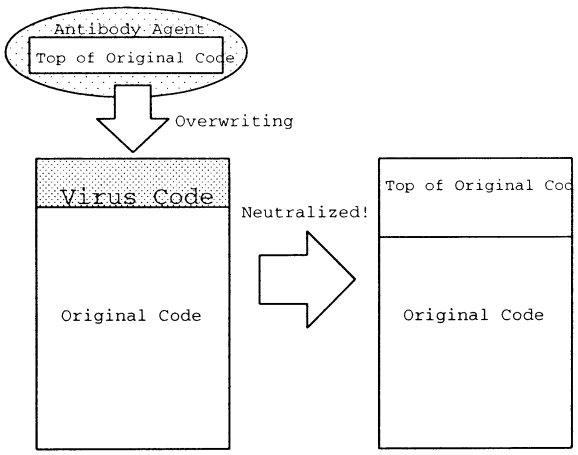


Figure 1: Neutralization by Antibody Agent

3.2 Recovery stage

Recovery from the infection can be done by a subsystem characterized as a distributed (over a network) and adaptive backup system. One may think that the backup system requires much diskspace for the backup files. However, each host has its own OS and system files on its disks except X-terminal and network computers. These system files that are distributed over network are redundant and can be used as backup with each other. That is, the backup files are already there. What is needed is a pointer to/from each file and a control mechanism to prevent a secondary infection in the copying process.

In this stage, two types of agents are used; killer agents and copy agents. The killer agents first remove the neutralized host program and activate the copy agents at the same time. The copy agents then send a request of copy with the self information corresponding to the file infected and netralized. Other copy agents who received the copy request, initiate the detection stage to check the requested file is clean. If it is clean, the copy is sent back to the original hosts. If the infectin is detected, the host make alarm and it cannot be a backup anymore.

There is one more type of agent called control agents which stays in the memory and control the entire process of the system. Antibody agents are placed in the bass between main memory and external memory to prevent virus program being loaded to the main memory by overwriting the head of the infected host program. As shown in Fig. 2, killer agents remove the neutralized file from the disk, and copy agent communicate with other copy agents staying at the disk of other hosts.

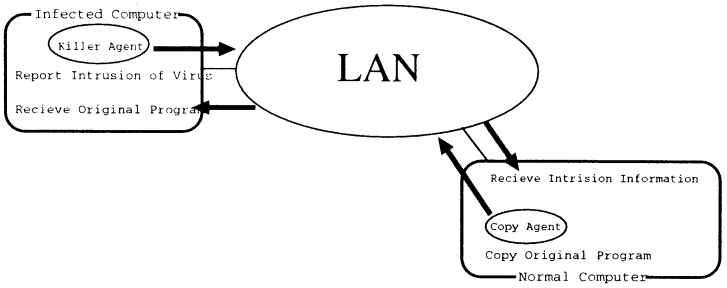


Figure 2: Damaged File Recovery by Killer Agent and Copy Agent

4 Experiments with Existing Computer Viruses

We have implemented a prototype of our system by JAVA (JDK1.1.3). The computer viruses used throughout these experiments are; `Scream3`,`Joker2`,`AIDS`, `Commander Bomber`, and `AP-605`. These are popular viruses representing different types.

4.1 Neutralization by antibody agents

The experiment is done on Pentium 133MHz CPU under Windows95. Firts, a system file of Windows95 is infected by each of the computer viruses. After the infection, our detection/neutralization subsystem and a commercial anti-virus program with scanner and checksum method are carried out. Then the infected program is executed to see if it is recovered or not. Our system successfully detects and neutralize all the infection. However, it cannot recover (neutralized though) from the infection of `AIDS` and `Commander Bomber`. `AIDS` alter the head of file more than the scope overwritten by antibody agents. `Commander Bomber` randomly separate its body and insert them into the host program. Therefore, the file cannot be normally executed after neutralization, however they already lost the execution priority. As for `Scream3`, although it can escape from the scanner and checksum method, it is detected and neutralied by our system. In summary, our detection/neutralization subsystem can detect, neutralize the infection, and further it can even

repair the infection whose alteration is less than "the self-information" kept.

4.2 Recovery by killer and copy agents

The experimental network consists of five computers: Sun ultra sparc 1; Compaq DESKTPRO; and Compaq ARMADA 4130T (x 3). We used several combinations of Solaris, Linux and Windows95 as OS. Since we do not have computer viruses agaist unix system, we used a file alteration similar to the viruses mentioned above. In this experiments with the specific environment, the infected file is successfully recovered all the case. Since we uses JAVA interpreter and did not optimize code at all. Performance is not yet seriously persued in the development of the current prototype.

The current prototype can deal with the file infection type virus only. However, by defining "macro programs" and boot programs in MBR, the method potentially can deal with the macro virus and the system region infection type. Performance in terms of additional cpu load and disk space required raise a future problem for practical use. Another important barrier for our system being used is psychological barrier to use the distributed and agents approach. Finally, the battle between virus programmer and anti-virus programmer forms a vicious circle. There is no complete anti-virus program as there is no complete virus program.

5 Conclusion

We proposed a distributed approach to computer virus detection/neutralization with autonomous and heterogeneous agents. Detection/neutralization of the infected system file can be done by keeping a few bytes of the head of the file as the "self" information. If the detection/neutralization subsystem fails to make recovery, the recovery subsystem which is an adaptive and distributed backup system is activated. The proposed method is implemented by JAVA, and tested in a small size network with five hosts, using the existing computer viruses.

Acknowledgments

This work has been supported in part by the SCAT (Support Center for Advanced Telecommunications Technology Research) Foundation.

References

[1] J. O. Kephart, G. B. Sorkin, D. M. Chess and S. R. White, "Fighting computer viruses," *Scientific American*, pages 88–96, November 1997.

[2] V. Bontchev, "Future trends in virus writing," *Proceedings of the 4th International Virus Bulletin Conference*, pages 65–82, 1994.

[3] F. Cohen, "Computer viruses, theory and experiments," *Proceeings of the 7th DOD/NBS Computer & Security Conference*, pages 22–35, September 1987.

[4] S. Forest et. al., "Self–nonself Discrimination in a computer," *Proc. of the 1996 IEEE Symp. on Comp. Security and Privacy*, pp 202–212, 16–18 May 1994.

[5] C. Pu et al., "A Specialization Toolkit to Increase the Diversity in Operating Systems," *IC-MAS Workshop on Immunity-Based Systems*, http://www.sys.tutkie.tut.ac.jp/~ishida/IMBS96proc.html

[6] Y. Ishida, "Fully Distributed Diagnosis by PDP Learning Algorithm: Towards Immune Network PDP Model," *Proc. of IJCNN 90*, San Diego, 1990.

[7] Y. Ishida and N. Adachi, "Active Noise Control by an Immune Algorithm" *Proc. Int. Conf. on Evol. Comp. 96*, pp. 150-153, 1996.

[8] Y. Ishida, "An Immune Network Approach to Sensor-Based Diagnosis by Self-Organization" *Complex Systems*, Vol. 10, No. 1, pp. 73-90, 1996.

[9] E. H. Spafford, "Computer virus–a form of artificial life?," *Artificial Life II*, pp. 727–745, Addison–Wesley, 1992.

[10] V. Bontchev, "Are "good" computer viruses still a bad idea?," *Proceedings of EICAR Conference*, pp. 25–47, 1994.

[11] V. Bontchev, "Possible virus attacks against integrity programs and how to prevent them," *Proceedings of the 2nd International Virus Bulletin Conference*, pp. 131–141, Sept. 1992.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

"Life Spacies": a genetic text-to-form editor on the internet

by Christa Sommerer (1) & Laurent Mignonneau (2)

(1) & (2) ATR Media Integration and Communications Research Laboratories
2-2 Amity Hikaridai, Seika-cho Soraku-gun, 61902 Kyoto, Japan
Tel: 81-77495-1426, Fax: 81-77495-1408
christa@mic.atr.co.jp, laurent@mic.atr.co.jp, http://www.mic.atr.co.jp/~christa

Keywords

Artificial Life, Genetic Algorithms, Internet, Virtual Reality, Interactivity, Interactive Art

Abstract

We are working to create interactive computer installations that integrate artificial life and real life through human-computer interaction [1] [2]. While exploring real-time interaction and evolutionary image processes, visitors to our interactive installations become essential parts of the systems by transmitting their individual behaviors to the work's image processing. Images in these installations are no longer static, prefixed and predictable but "life-like systems" in themselves, reflecting minute changes in the viewer's interactions with the evolutionary image processes.

1. Introduction

Based on the principles of open-ended design, we have developed an interactive computer installation called "Life Spacies." "Life Spacies" was produced for the ICC InterCommunication Museum in Tokyo as part of its permanent collection [3]. It is an interaction and communication environment where remotely located visitors on the internet (i.e., global environment) and the on-site visitors to the installation at the ICC Museum in Tokyo (i.e., local environment) can interact with each other through artificial creatures.
Artificial creatures are created by on-line participants through writing email messages to the "Life Spacies" web page. Each text message provides the genetic code for a creature, and a text-to-form editor developed by Laurent Mignonneau allows us to translate text into form.
The following paper describes this text-to-form editor in more detail and explains the interactions between creatures and visitors and those among creatures in the "Life Spacies" interaction environment.

2. From Text to Form on the Internet

The "Life Spacies" web page (Fig. 1) allows people throughout the world to interact with the system. By simply typing and sending an email message to the "Life Spacies" web site (http://www.ntticc.or.jp/~lifespacies), one can create one's own artificial creature: this creature starts to live in the interaction environment at the ICC Museum as soon as the message is sent.

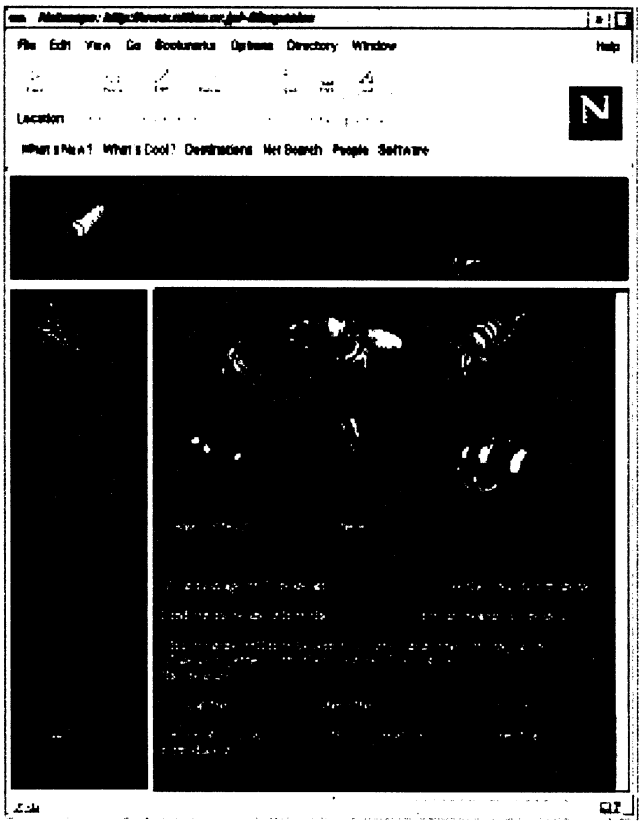


Fig. 1 "Life Spacies" web site

We developed a special text-to-form editor that enables us to translate text into genetic code.

3. Text-to-Form Editor

The text-to-form editor is based on the idea of linking the characters and syntax of a text to specific parameters in the creature's design. The default form of a creature is a body made up by a sphere consisting of 100 vertices, 10 rings

with 10 vertices each. All vertices can be modified in x, y and z axes to stretch the sphere and create new body forms. Several bodies can be attached to each other provided that their attachment point is located on the x-axis. If the attachment point is not on the x axis, a limb is created instead of a body; this limb is copied and the copy is attached at a position symmetric to the original position. Figure 2 show a creature with two spheres as bodies and one pair of limbs.

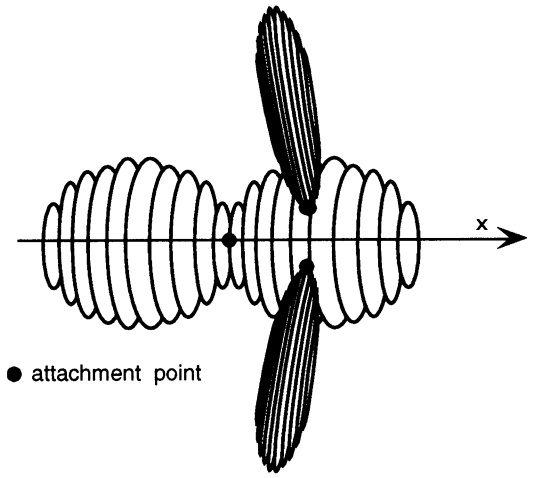


Fig. 2 Creature with two bodies and one pair of limbs

According to the sequencing of the characters in the text, the parameters of x, y and z for each of the 100 vertices can be stretched and scaled, the color values and texture values for each body and limb can be modified, the number of bodies and limbs can be changed and new locations for attachment points of bodies and limbs can be created. Since each of the vertex parameters is changeable and all of the bodies and limbs can be changed as well, about 50 different functions for the creature's design parameters are available. The design functions are subsumed in a design function table (Fig. 3).

function1 stretch default body/limbs in x
function2 stretch default body/limbs in y
function3 stretch default body/limbs in z
function4 set the next stretch function to global
function5 set the next stretch function to a vertex point
function6 set the next stretch function to a ring
function7 create a new location for an attachment point
function8 copy a new location for an attachment point
function9 compose a new texture for body/limbs
function10 copy texture of body/limbs
function11 change parameters of RED in body/limbs texture
function12 change parameters of GREEN in body/limbs texture
function13 change parameters of BLUE in body/limbs texture
function14 change patterns of body/limbs texture
function15 exchange positions of bodies/limbs
function16 copy body/limbs
function17 create a new body/limbs
function18 add or replace some of the above functions
function19 randomize the next parameters
function19 copy parts of the previous operation
function20 modify life span (default is 24 hours)
function21 add the new parameter to previous parameter
function22 ignore the current parameter
function23 ignore the next parameter
function24 replace the previous parameter by new parameter
.....
function50

Fig. 3 Design function table

Next, in translating the characters of the text message into these design function values, we first assign an ASCII value for each character. This is done according to the standard ASCII table shown in Figure 4.

33 !	34 "	35 #	36 $	37 %	38 &	39 '	
40 (	41)	42 *	43 +	44 ,	45 -	46 .	47 /
48 0	49 1	50 2	51 3	52 4	53 5	54 6	55 7
56 8	57 9	58 :	59 ;	60 <	61 =	62 >	63 ?
64 @	65 A	66 B	67 C	68 D	69 E	70 F	71 G
72 H	73 I	74 J	75 K	76 L	77 M	78 N	79 O
80 P	81 Q	82 R	83 S	84 T	85 U	86 V	87 W
88 X	89 Y	90 Z	91 [	92 \	93]	94 ^	95 _
96 `	97 a	98 b	99 c	100 d	101 e	102 f	103 g
104 h	105 i	106 j	107 k	108 l	109 m	110 n	111 o
112 p	113 q	114 r	115 s	116 t	117 u	118 v	119 w
120 x	121 y	122 z	123 {	124 |	125 }	126 ~	

Fig. 4 ASCII table

We see that each character refers to an integer. We can now proceed by assigning this value to a random seed function *rseed*. In our text example from Figure 4, *T* of *This* has the ASCII value 84, hence the assigned random seed function for *T* becomes *rseed(84)*. This random seed function now defines an infinite sequence of linearly distributed random numbers with a floating point precision of 4 bytes (float values are between 0.0 and 1.0). These random numbers for the first character of the word *This* will become the actual values for the modification parameters in the design function table. Note that the random number we use is a so-called "pseudo random," generated by an algorithm with 48 bit precision, meaning that if the same *rseed* is called once more, the same sequence of linearly distributed random numbers will be called. Which of the design functions in the design function table are actually updated is determined by the following characters of the text, i.e., *his*; we then assign their ASCII values (104 for *h*, 105 for *i*, 115 for *s* ...), which again provide us with random seed functions *rseed(104)*, *rseed(105)*, *rseed(115)*. These random seed functions are then used to update and modify the corresponding design functions in the design function look-up table, between the design function1 and function50. For example, by multiplying the first random number of *rseed(104)* by 10, we get an integer, which assigns the amount of functions that will be updated. Which of the 50 functions are precisely updated is decided by the following random numbers of *rseed(104)* (as there are 50 different functions available, the following floats are multiplied by 50 to create integers). Figure 5 shows in detail how the entire assignment of random numbers to design functions

operates. As mentioned above, the actual float values for the update parameters come from the random seed function of the first character of the word, *rseed(84)*. An example of the entire procedure is given in Figure 5.

Example word: *This*

T => *rseed(84)* => {0.36784, 0.553688, 0.100701,...}
(actual values for the update parameters)

h => *rseed(104)* => {0.52244, 0.67612, 0.90101,...}
/

0.52244 * 10 => get integer 5 => 5 different functions are called within design function table

0.67612 * 50 => get integer 33 => function 33 within design function table will be updated by value 0.36784 from 1. value of *rseed(84)*

0.90101 * 50 => get integer 45 => function 45 within design function table will be updated by value 0.553688 from 2. value *rseed(84)*

..... until 5. value

Fig. 5 Example of assignment between random functions and design functions

As explained earlier, the basic "module" is a sphere, with a default color white and no texture. When messages are sent, the incoming text modifies and "sculpts" this default module by changing its form, size, color, texture, number of bodies/limbs, copying parts and so forth. Depending on the complexity of the text, the body and limbs of the creature become increasingly shaped, modulated and varied. As there is usually great variation among the texts sent by different people, the creatures themselves also vary greatly in appearance, thus providing a personal creature for each author.

Figure 6 shows an example of a short and simple email message sent to the "Life Spacies" web site.

Date: Sun, 01 Nov 1998 13:14:32 +0900
From: Christa Sommerer <christa@mic.atr.co.jp>
To: life@lc.ntticc.or.jp
CC: christa@mic.atr.co.jp
Subject: test creature1

This is a test creature.

Fig. 6 Example of email message to "Life Spacies"

4. Picture of the Creature

As soon as this message is sent to the server in Tokyo, the creature starts to live in its virtual environment and the author of the text receives a picture of his or her creature in return.

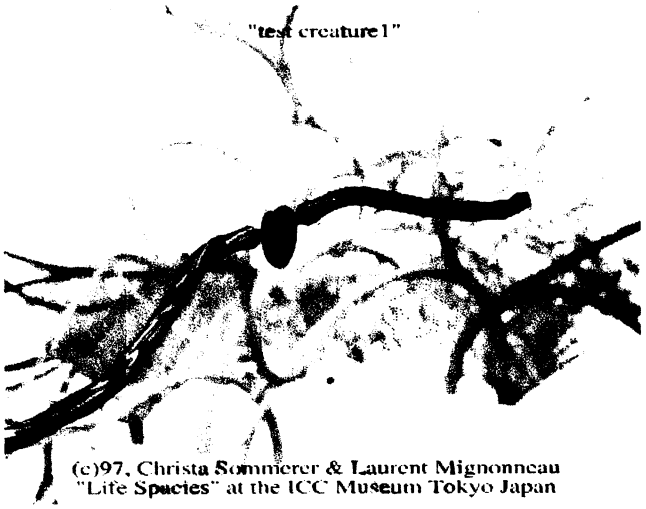


Fig. 7 Creature created by email in Fig. 6

Figure 7 shows an image of the creature created by the text message of Fig. 6. Because the text message was rather short, the corresponding creature consists just of one body and one pair of limbs, similar to the default case but with long limbs and a heart-shaped body.

5. Variations in Creature's Design

By only changing the first character of each word, a different random seed is chosen for the following characters of the word and, consequently, the design for body and limbs will change. Figures 8 and 9 show the effects of changing the original message of Figure 6 by modifying the *H* into *F*, the *T* into *R*, *i* into *o*, *t* into *n* and *c* into *g*.

Date: Sun, 01 Nov 1998 13:15:07 +0900
From: Christa Sommerer <christa@mic.atr.co.jp>
To: life@lc.ntticc.or.jp
CC: christa@mic.atr.co.jp
Subject: test creature1var2

Rhis os i next greature.

Fig. 8 Modified email message

We see that the new creature in Fig. 9 still consists of one body and one pair of limbs, but its form, size, orientation and color of body and limbs have changed significantly.

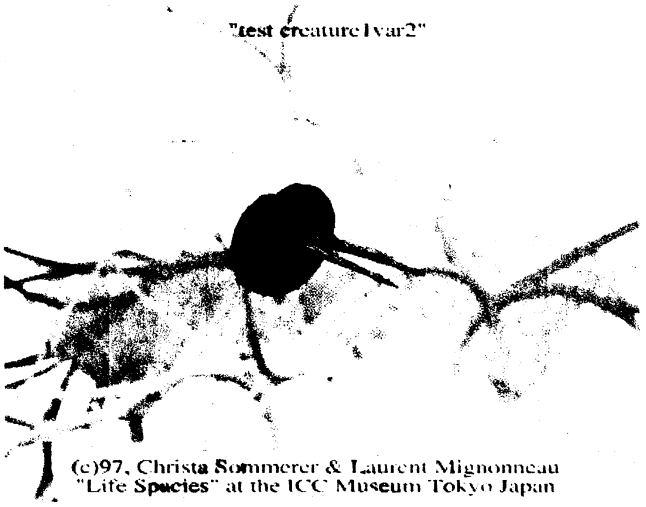


Fig. 9 Creature created by modified email message

When more complex messages with more characters, words and varied syntax are sent, more elaborate creatures with more bodies, limbs and variation in body form, texture, size and color can be created. Figure 11 shows an example and Fig. 10 is the corresponding text message.

Date: Sun, 01 Nov 1998 13:20:32 +0900
From: Christa Sommerer <christa@mic.atr.co.jp>
To: life@lc.ntticc.or.jp
CC: christa@mic.atr.co.jp
Subject: example #4

this is not a sentence, it is a creature, it is now in Tokyo, where it lives.
it is a creature, this is not a sentence, where it lives, it is now in Tokyo.
it is now in Tokyo, this is not a sentence, it is a creature, where it lives.
where it lives, it is a creature, it is now in Tokyo, this is not a sentence.

Fig. 10 Complex email message

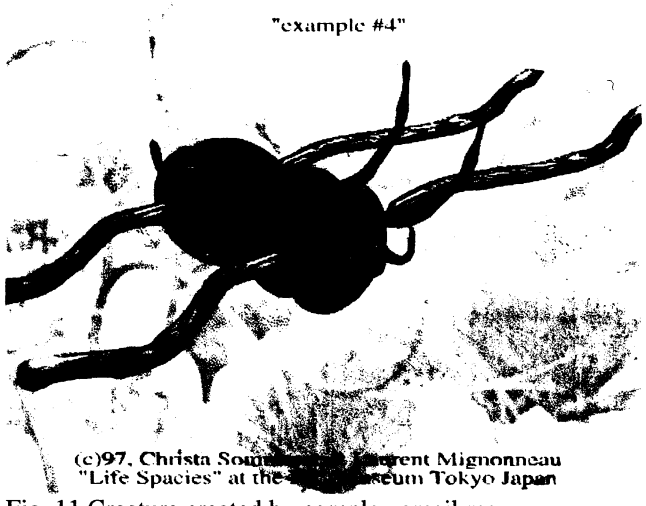
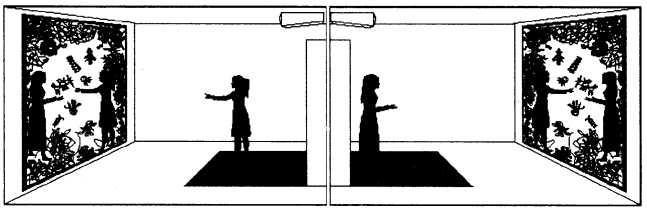


Fig. 11 Creature created by complex email message

7. Interaction Setup

The interaction setup consists of two independent interaction sites (Fig. 12) that are linked together via a data line, allowing visitors at remote locations to be displayed and interact in the same virtual three-dimensional space. The system setup is based on earlier interactive installations called "Trans Plant" [4] and "MIC Exploration Space"[5].


Fig. 12 Life Spacies Interaction Setup

8. Interaction between Visitors and Creatures On-site

On-site visitors can directly interact with the creatures through touching and catching them. Once a creature is caught by the visitor, it will clone itself. However, if two remotely located people are in the same virtual space, they can each catch a creature with their hands, which causes these two creatures to mate and to create an offspring by genetically exchanging the parents' code.

9.1. Cloning Creatures

If the visitor catches a creature it makes a perfect copy, or clone, of itself. The creatures are basically shy, and one needs to look for them carefully because they hide in the branches of vegetation.

9.2. Mating Creatures

When the two remotely located visitors each catch a creature, these two creatures mate and create an offspring, a child creature. In this case, the offspring inherits the genetic code of the parent creatures; this is done through cross-over of the parents' codes and application of minimal mutation. Cross-over can take place at any part of the genetic string (i.e., text) of the creatures; the location and length of the cross-over is decided at random, but it is adapted to the length of the genetic string (i.e., text) of the creatures. Figure 13 shows an example of a genetic exchange through cross-over and mutation.

Parent creatures (1) and (2), child creature (3);
| .... indicates the area of cross over;
^ .... indicates the location of mutation;

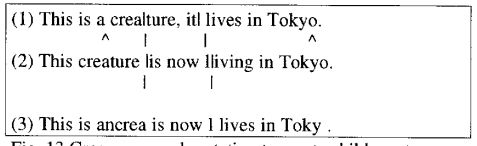


Fig. 13 Cross-over and mutation to create child creature

10. Curriculum Vitae of Creature

A creature's default life span is 24 hours, but as the life span is also a function from the design function table (Fig. 3), it will be updated and changed through the values of the specific characters in the text. When the creature has died, a report is given to it's author, telling him or her how long the creature lived and how many children and clones it produced. Figure 14 shows the Curriculum Vitae of the creature in Fig. 11.

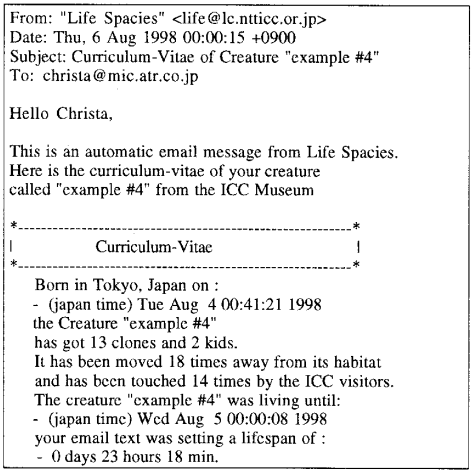


Fig. 14 Curriculum Vitae of creature in Fig. 11

11. Conclusions

"Life Spacies" is a system where interaction and exchange happens between real life and artificial life on human-human, human-creature and creature-creature levels. Aided by our genetic text-to-form editor, users on the Internet can create artificial creatures through writing text messages to the "Life Spacies" web site; additionally on-site visitors to the installation influence the creatures' reproduction by touching them with their hands and thus promote the propagation of specific gene pools of creatures in the "Life Spacies" environment.

References:

[1] Sommerer C and Mignonneau L (1998), Art as a Living System. In: Art @ Science. C. Sommerer and L. Mignonneau (eds), Springer Verlag, Vienna New York, pp 148-161

[2] Sommerer C and Mignonneau L (1997), Interacting with Artificial Life: A-Volve. In: Complexity Journal. H. Morowitz and J. Casti (eds), Wiley, New York,Vol. 2, No. 6, pp 13-21

[3] Sommerer C and Mignonneau L (1997), Life Spacies. In: ICC Concept Book, NTT-ICC, Tokyo, pp. 96 -101

[4] Sommerer C and Mignonneau L (1995), Trans Plant. In: Imagination. T. Moriyama (ed), Tokyo Metropolitan Museum of Photography, Chapter 2

[5] Sommerer C and Mignonneau L (1996), MIC Exploration Space. In: Siggraph'96 Visual Proceedings, ACM Siggraph, New York, p. 17

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Learning in Software Driven Neural Networks with Temporal Coding and Functional Connectivity

Masataka Watanabe †,††*and Kazuyuki Aihara* †,††
Email:watanabe@sat.t.u-tokyo.ac.jp

†Dept. of Department of Mathematical Engineering and Information Physics, Tokyo University
7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan
††CREST, Japan Science and Technology Corporation (JST)
4-1-8 Honmachi, Kawaguchi, Saitama, Japan

ABSTRACT

Functional connectivity is dynamic neural interactions specific to temporal pulse coding. Using functional connectivity, we introduce a "software driven neural network" which is capable of switching its pattern classification properties by change of "software input". We also give a learning algorithm for our model which makes it possible to learn the classification properties.

KEYWORDS: temporal pulse coding, functional connectivity, software driven neural networks, learning

1. Introduction

Over the past few decades, researches of modern scientists have thrown new light on the nature's most highly complicated system, the brain. It is true that much fascinating truth of the brain has been revealed due to the current progress in experimental technique and computer simulation. Although, a very fundamental problem remains an open question:*what is the carrier of information in the brain?*

It is quite obvious that the brain makes use of spikes transmitted among neurons through synaptic connections for information processing. The question is what constitutes information in these spike trains. Is it the mean firing rate or the timing of spikes? This is a key question that must be solved to understand the information processing and the knowledge representation of the brain.

The traditional view to this question is that mean spike rate in a psychological time scale of hundreds of milliseconds is the carrier of information. Many researchers came to believe this hypothesis due to an experimental evidence that the torque generated in muscles was proportional to the spiking rate in motor systems (Robinson[1]). Most physiological experiments and neural network models are based on the rate coding hypothesis.

However, the possibility of temporal pulse coding, which assumes the spike timing as the carrier of information, has been recently brought to light by Malsburg[2] . His theoretical viewpoint was that timing of spikes plays an essential role in encoding, representing, and processing information and knowledge in the brain. This pioneering work was followed by physiologists seeking for proof of temporal pulse coding in the actual brain. Since the most significant difference of rate coding and temporal pulse coding is whether the neuron functions as integrators or coincidence detectors, considerable numbers of attempts were made to clarify this question. Softky and Koch[3] reported that the inter spike intervals(ISI) of cortical pyramidal cells are highly variable, which is considered as a circumstantial evidence of these neurons to function as coincidence detectors. Although a counter-argument was made by Shadlen and Newsome[4] that the irregularity of the ISI may be explained by balance of excitatory and inhibitory incident spikes, Softky[5] came up with an attractive explanation: the effective coincidence detector hypothesis. This hypothesis claims that the presence of balanced excitatory and inhibitory inputs results to the functioning of an integrator neuron as a coincidence detector.

Apart from the question of the function of cortical neurons, Vaadia, Aertsen, Abeles and their colleagues have observed dynamical modulation of temporal correlations in a short time-scale of the order of 100 ms in a task-related manner(Vaadia[6] Aertsen[7]). This result is considered as a more direct proof of temporal pulse coding in cortical information processing. There are still other experimental work which suggest the plausibility of temporal pulse coding in the brain (Sakurai[8], Tsuda[9]). From the theoretical side, Maass have showed that network of spiking neurons are computationally more powerful than networks composed of McCulloch Pitts neurons (Mass[10]).

While the heated debate of whether the brain is using temporal coding or rate coding or even both is pursued, we look at this problem from another view point. We consider what new tools can temporal coding provide to neural network models. Our opinion is that one of the keys of temporal coding is the dynamic neuronal interaction, namely, "functional connectivity".

In this paper, we first give a brief description of functional connectivity in contrast with classical synaptic connectivity in rate coding and next move on to the central to this issue, *what it can do.* Here, we come up with a new concept, that is to say, neural networks which are capable of switching its functions by "software" (Watanabe[11]). Of course the essential to this model is functional connectivity. We also introduce a learning rule for our new model.

2. Functional connectivity

To begin with, the most significant difference between rate coding and temporal pulse coding is based upon the behavior of neurons(figure(1)). The decay constants characterize the behavior of the neurons. A neuron in rate coding has a large decay constant, so that the neuron acts as an integrator of incident pulses. On the other hand, the decay constant of a neuron with temporal pulse coding is very small, and the neuron functions as a coincidence detector of input pulses. In this case, the timings of input pulses become a significant factor for information processing.

Now that we are ready, we consider the difference of interaction between neurons in rate coding and temporal pulse coding, namely, the classical synaptic connectivity and func-

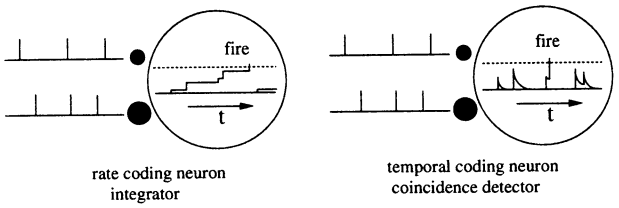


Figure 1: Neuronal interaction in rate coding and temporal pulse coding

tional connectivity. In the case of rate coding, pulses from a pre-neuron give almost equal influence to the post-neuron irrelevant to input timing; thus, the neuronal interaction in rate coding is static. In contrast, interaction in temporal pulse coding is dynamic because the influence of a single pulse to a neuron depends on input timing with other incident pulses due to small decay time constants. We can say that the strength of connection is determined not only by the synaptic efficacy but also the ongoing network dynamics. Such dynamic connection in temporal pulse coding is called functional connectivity(Malsburg[2], Aertsen[7] Fujii[12]).

3. Software driven neural networks

As we noted in the previous section, in order to make use of functional connectivity we must use the property that it is dynamic. We came up with a simple idea; a neural network with multiple functions.

Most neural network models so far studied are single-functional, that is to say, the objective is implemented in the neural hardware and once the hardware is constructed either by design or by learning and self-organization, it can be only used for such a particular purpose although it can still hold plasticity for further learning and self-organization.

This stubbornness of conventional neural network models is due to the fact that processing of input data mainly relies on the fixed connections among neurons. Surely synaptic connections can be plastic, but this plasticity is only in the long term. Generally the time scale of learning and self-organization is much longer than that of processing. So as long as the actual processing of input data is of interest, we can assume that values of synaptic connections are fixed.

Using functional connectivity we may exceed the limits of neural networks with classical synaptic connectivity. If we can change the excitation dynamics of the network, hence change the functional connectivity, the network may process multi-functionally. We may consider the input to the network which controls the basic dynamics as "software". In the next section, we describe our neuron and network model in detail.

4. Network model

The neuron model used here is a continuous time and point process model with deterministic dynamics. A neuron adds up effects of input pulses as activation $a(t)$ which decays exponentially with a time constant τ. When $a(t)$ becomes larger than or equal to the sum of the firing threshold θ and the global negative feedback value $r(t)$, namely $a(t) \geq \theta + r(t)$, then the neuron fires with time delay $g(h(t))$, where $h(t) = a(t) - (\theta + r(t))$ and $g(u) = \alpha/(\beta + u)$. This time delay described by the function g models a relationship between strength of suprathreshold stimulation and latency of action potential, or the time difference between the activation to exceed the threshold and the generation of the action potential at an axon hillock. The higher the value of $h(t)$ is, the faster is the firing.

We mutually connect the neurons described above with finite delays to construct a network. The global negative feedback value is calculated by summing up output pulses from all neurons with a decay time constant τ_g and assumed to change only the level of the threshold. The network model is described as follows,

$$\tau da_i(t)/dt = -a_i(t) + S^{ext} \sum_n \delta(t - t^p_{i,n}) + \sum_{j=1}^{N} w_{ij} x_j(t - d_{ij}), \tag{1}$$

$$\tau_g dr(t)/dt = -r(t) + R \sum_{j=1}^{N} x_j(t) \tag{2}$$

$$\begin{aligned} &If \qquad h(t) \equiv a_i(t) - (\theta + r(t)) \geq 0 \\ &\quad x_i(t') = \delta(t' - g(h(t)) - t) \;\; and \;\; a_i(t') = 0 \\ &otherwise \\ &\quad x_i(t) = 0, \end{aligned}$$

where S^{ext} is the strength of the external input pulse, R the strength of each global negative feedback weight, w_{ij} and d_{ij} are the synaptic weight and delay for pulse propagation from neuron j to neuron i, respectively, and N the number of neurons in the network. All synaptic weights are set to 1 except self-connections $w_{ii} = 0 \;\; (i = 1, ..., n)$, while delays d_{ij} are assumed to be randomly determined according to a normal distribution with mean $\bar{d}$ and variance $< d >$. Moreover, $t^p_{i,n}$ denotes the time for the n th external pulse arriving at the i th neuron.

Turning now to software input, it plays an important role in our network model. The software input determines the dynamics of the network and hence the functional connectivity. In this study, the software input correspond to a temporal spike train given to particular neurons (3 out of 64 neurons) in the network. The connection weights to these neurons from the software input is strong enough so that it can fire these neurons by itself. Moreover, the spike train is always given to the same neurons and only the temporal pattern of spikes is changed for the sake of simplicity.

Next, the pattern input to the network is in the form of spatial suppression of neurons. We choose the neurons to suppress according to the given spatial input pattern as shown in figure (2). This suppression of neurons is equivalent to receiving high frequency pulses through inhibitory connections. The meaning of suppressing neurons is taken up in the section of Discussion.

The output is given in the form of a spatio-temporal firing pattern of neurons in the network. We can express this output by plotting firing time to each neuron index as shown in figure (2).

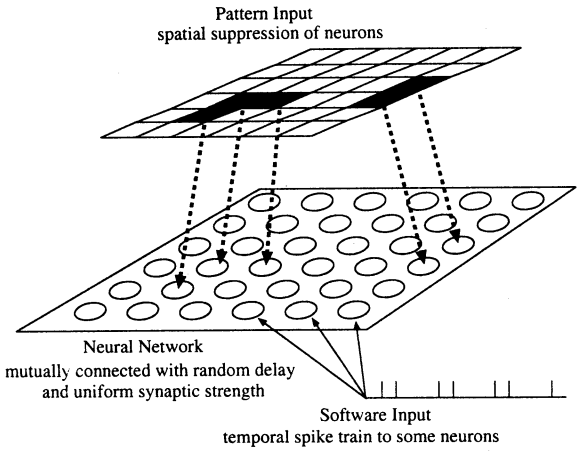


Figure 2: Neural network model

5. Learning rule

Here, we introduce a learning algorythm for our software driven neural network. The main idea of this algorythm is to adjust the pulse flow, in other words, the chain of functional connectivity so that it will flow through neurons which correspond to the differents parts of the patterns which we want to categorize. We assume a local synaptic delay changing rule whenever a neuron fires as in the following.

$$\Delta d_{i,j} = \delta(t^p_{i,l} - t^p_{i,l-1}), \Delta d_{i,k} = \delta(t^p_{i,l-1} - t^p_{i,l}), \qquad (3)$$

where δ is the learning constant and $t^p_{i,l}$ is the pulse which directly triggered the firing of the neuron. Index j and k denote the pre-neurons which sent out the pulses $t^p_{i,l-1}$ and $t^p_{i,l}$ respectively. This local rule works so that whenever neurons fire periodically, it becomes more likely to fire in the next cycle, since the pulse interval of incident pulses which triggered the neuronal firing decreases.

On the other hand, when a neuron raised its internal state large enough to fire but could not fire due to spatial depression, we apply a delay changing rule as in the following.

$$\Delta d_{i,j} = -\delta(t^p_{i,l} - t^p_{i,l-1}), \Delta d_{i,k} = -\delta(t^p_{i,l-1} - t^p_{i,l}), \qquad (4)$$

The above rule functions to kick out neuron which receive spatial depression due to pattern input from the periodic neuronal firing.

We apply the above two local learning rules while inputting the common parts of the spatial input pattern. As a result, a pulse stream which flows through the non-common parts of the spatial input pattern is constructed and therefore it becomes possible to categorize the patterns looking only at the non common parts.

6. Simulation Results

6.1. Spatio-temporal output of the network without pattern input

Here we give the output of the proposed model without pattern input and regular spike train as software input. Figure (3) is a raster plot which indicate when and what neurons fired. The neuronal firings due to software input is shown in the lower part of the diagram, that is to say, neuron 1,2 and 3 firing periodically. We can also see that the dynamics of the whole network is also periodic. We confirmed in our previous work(Watanabe[13]) that depending on neuronal parameters, the dynamics of the whole network becomes also chaotic.

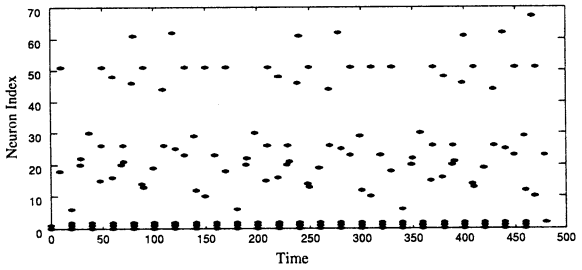


Figure 3: Spatio-temporal firing of the network

6.2. Dual Pattern Classifying Properties by Two Software Inputs

We will see what happens when we give a pattern input. The pattern input is given as the spatial suppression of neurons. Each cell in figure (4) represent the result for a single input pattern where a input pattern is the sum of two patterns in the X axis and the Y axis. The grey scale indicate the result of classification, namely, same grey scale for same spatio-temporal output of the network for each software input. We can see that the pattern classification property differ between the two software inputs.

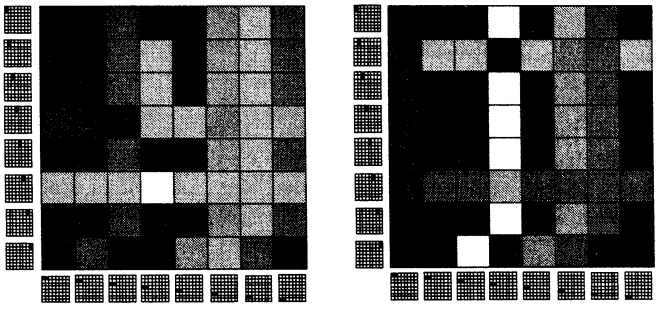


Figure 4: Pattern classification properties of the two software inputs

6.3. Learning of Pattern Classification

In this section we will look at the results of learning of pattern classfications. Figure (5) gives the results before applying the learning algorythm. This figure indicates that patterns A and

D, B and C are categorized as one. On the other hand, after learning (Fig.(6)), patterns A and B, C and D are categorized as one. This is due to the learning algorythm where we gave inhibitory input to neurons corresponding to the striped part in the diagram at the same time. A pulse flow which flows between these neurons will be organized, and therefore, the network dynamics will not be affected by these neurons.

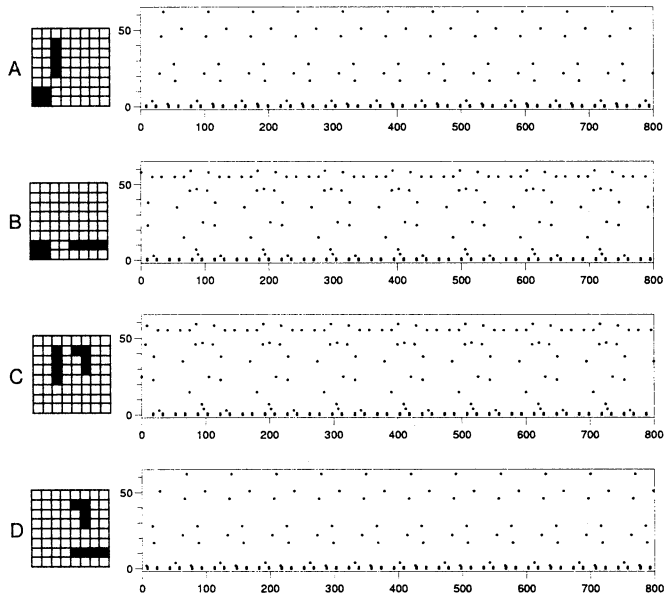


Figure 5: Network output before learning

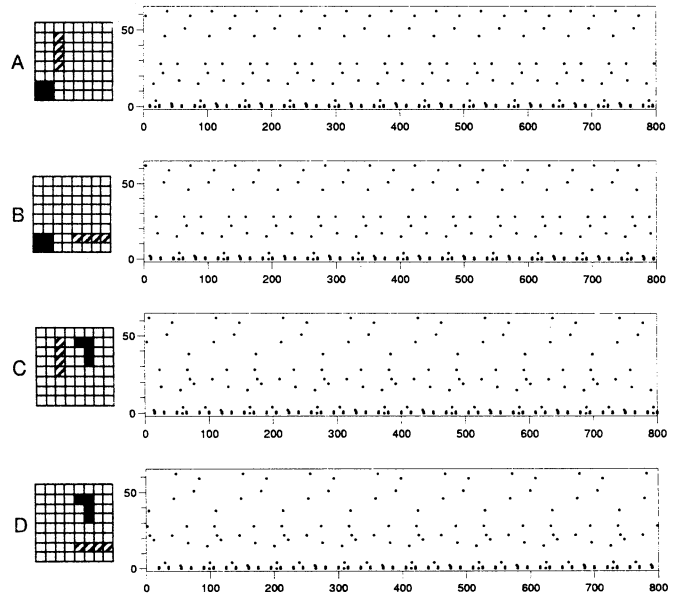


Figure 6: Network output after learning

7. DISCUSSION

We have proposed a neural network model which is capable of altering its pattern classification properties by change of so called "software input". The key to this model is functional connectivity which is dynamic neuronal interaction peculiar to temporal pulse coding. We have also introduced a learning algorythm which is capable of changing the pattern classification properties.

The most important subject to our model which should be taken up is biological plausibility. In our current simulations, neurons fire with only two coincident pulses. Therefore our model is very critical to noise. We think that it is possible to overcome this problem by assuming neurons as "bunch of neurons" connected by syn-fire chains.

References

[1] Robinson, D.A.(1975), *Oculomotor control signals – Basic mechanisms of ocular mobility and their clinical implications* –, Pergamon Press .

[2] Von der Malsburg, C.(1987), *Synaptic plasticity as basis of brain organization – The neural and molecular basis of learning* –, John Wiley and Sons.

[3] Softky, W. R. and Koch, C.(1993), "The highly irregular firing of cortical cells is inconsistent with temporal integration of random EPSPs," *J. Neurosci*, **Vol. 13**, pp.334–350.

[4] Shadlen, M.N. and Newsome, W.T.(1994), "Noise neural codes and cortical organization," *Current Opinion in Neurobiology*, **Vol. 4**, pp.569–579.

[5] Softky, W.R., "Simple codes versus efficient codes, " *Current Opinion in Neurobiology*, **Vol. 5**, pp.239 – 257 (1995).

[6] Vaadia, E., Haalman, I., Abeles, M., Bergman, H., Prut, Y., Slovin, H., Aertsen, A.(1995), "Dynamics of neuronal interactions in monkey cortex in relation to behavioral events," *Nature*, **Vol.373**, pp.515–518 .

[7] Aertsen, A.M.H.J. and Gertstein, G. (1991), *Dynamics aspects of neuronal cooperativity: fast stimulus-locked modulations of effective connectivity – Neuronal cooperativity* –, Springer-Verlag

[8] Sakurai, Y. (1993), "Dependence of functional synaptic connections of hippocampal and neocortical neurons on types of memory," *Neuroscience Letters*, **Vol.158**, pp.181–184.

[9] Tsuda, I.(1992), "Dynamic link of memory - chaotic memory map in nonequilibrium neural networks," *Neural Networks*, **Vol.5**, pp.313–326 .

[10] Maass, W.(1997), "Networks of spiking neurons: the third generation of neural network models," *Neural Networks*, **Vol.10**, 9, pp.1659–1671 .

[11] Watanabe, M., Aihara, K. and Kondo, S.(1998), "A dynamical neural network with temporal coding and functional connectivity," *Biological Cybernetics*, **vol.78**, pp.87–93 .

[12] Fujii, H., Ito, H., Aihara, K., Ichinose, N., Tsukada, M.(1996), "Dynamical cell assembly hypothesis - theoretical possibility of spatio-temporal coding in the cortex," *Neural Networks*, **vol.9**, pp.1303–1350 .

[13] Watanabe, M. and Aihara, K.(1997)," Chaos in neural networks composed of coincidence detector neurons," *Neural Networks*, **vol.10**, 8, pp.1353–1359 .

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Retrieval Characteristcs of Associative Chaotic Neural Networks with Weighted Pattern Storage

Masaharu Adachi
Department of Electronic Engineering
College of Engineering
Tokyo Denki University
Chiyoda-ku, Tokyo 101-8457, Japan

Kazuyuki Aihara
Department of Mathematical Engineering
and Information Physics
The University of Tokyo
Bunkyo-ku, Tokyo 113, Japan

CREST, Japan Science and Technology Corporation(JST)
4-1-8 Hon-cho, Kawaguchi, Saitama 332, Japan

Abstract

Retrieval characteristics of an associative chaotic neural network with weighted pattern storage are analyzed in the present paper. Four binary patterns are stored as basal memories to the network. In the previous studies, the four stored patterns are used equally to determine the synaptic weights, which store the information about the stored patterns. On the other hand, in this paper, we investigate the case when one of the stored pattern is stored weaker than the other stored pattern. The retrieval characteristics of the network in such a case are analyzed.

Keywords: Chaos, Associative neural network, Weighted auto-associative matrix

1 Introduction

It has been reported that associative neural networks which composed of chaotic neuron models [1] show non-periodic retrieval of the stored patterns [2][3][4]. Such non-periodic pattern retrieval is called dynamical association. The dynamical association of the associative chaotic neural network is caused by two contradicting forces of feedback inputs to each constituent neuron in the network from other constituent neurons and the refractoriness of the constituent neuron. The dynamical association is different from the associations realized in the conventional associative networks where retrieval in the conventional ones means retrieving one of the stored patterns from their perturbed initial patterns [5][6][7] [8][9].

In the previous study on the characteristics of the associative chaotic neural networks, the synaptic weights, which are used to embed the stored patterns to the network, are determined by the auto-associative matrix that is also used in the conventional associative networks. In the method of determining the synaptic weights, all the stored patterns affect equally to the synaptic weights because the auto-associative matrix is just calculating the correlations among the stored patterns. Recently, a weighted auto-associative matrix is used in order to study the chaotic itinerancy phenomenon of the associative chaotic neural network [10]. In Ref.[10], an associative chaotic neural network composed of four neurons with two orthogonal stored patterns is analyzed. The network is suitable to study the bifurcation structure, however it is not suitable to study about the retrieval ability of the stored patterns, therefore, in this paper, we investigate retrieval characteristics of an associative chaotic neural networks composed of one hundred chaotic neurons with four patterns stored with the weighted auto-associative matrix.

2 Associative Chaotic Neural Network Model

The associative chaotic neural network model consists of chaotic neuron models. The chaotic neuron model has sigmoid output function and self feedback connections realizing the refractoriness that is observed in real neurons. Due to these features, single neuron model exhibits deterministic chaos with some parameters [1]. The network model in the present paper is described by the following equations:

$$x_i(t+1) = f\{\eta_i(t+1) + \zeta_i(t+1)\},$$
$$\eta_i(t+1) = k_f \eta_i(t) + \sum_{j=1}^{100} w_{ij} x_j(t),$$

$$\zeta_i(t+1) = k_r\zeta_i(t) - \alpha x_i(t) + a_i$$

where $x_i(t)$ denotes output of the ith neuron at discrete-time t; $\eta_i(t)$ and $\zeta_i(t)$ denote internal states for feedback inputs from the constituent neurons and for refractoriness, respectively; k_f and k_r are the decay parameters for the feedback inputs and the refractoriness, respectively; w_{ij} and a_i denote the synaptic weights from the jth neuron to the ith neuron and the sum of the threshold and the temporally constant external inputs to the ith neuron, respectively. The output function of the neuron is denoted by f; in the present paper, we use the logistic function represented by

$$f(y) = \frac{1}{1+\exp(-y/\varepsilon)}, \qquad (1)$$

where ε is a parameter for adjusting the steepness of the function [1] [3].

2.1 Conventional auto-associative matrix

In the previous studies [2][3][4][11][12], the synaptic weights of the associative chaotic neural networks are determined by the following auto-associative matrix [13] of four 100-dimensional binary vector patterns to be stored:

$$w_{ij} = \sum_{p=1}^{4}(x_i^{(p)} - \overline{x})(x_j^{(p)} - \overline{x}) \qquad (2)$$

where $x_i^{(p)}$ is the ith component of the pth stored pattern and $\overline{x}$ is a spatially averaged value of the stored patterns. Using the auto-associative matrix is the standard method for determining the synaptic weights of the conventional associative networks [5] [6] [7] [8] [9].

2.2 Weighted auto-associative matrix

In the present paper, we investigate the retrieval characteristics of the associative chaotic neural network with an weighted auto-associative matrix represented as follows [10]:

$$w_{ij} = \sum_{p=1}^{4}\omega^{(p)}(x_i^{(p)} - \overline{x})(x_j^{(p)} - \overline{x}) \qquad (3)$$

where $\omega^{(p)}$ denotes weight of the pth stored pattern. In general, $\omega^{(p)}$ may take a value between zero and one. When $\omega^{(p)} = 1$ for every pattern ($p = 1, 2, 3, 4$) Eq.(3) is equivalent to Eq.(2) (i.e., the conventional auto-associative matrix). In the present paper, we consider the case where only one of the four stored patterns is weakly weighted than the other stored patterns, namely, only one of $\omega^{(p*)}$ among $p = 1, 2, 3, 4$ is set to a value less than one and the other $\omega^{(p)}$'s for $p \neq p*$ are set to one. For example when we would like to store the first pattern weaker than the other stored patterns, $\omega^{(p)}$'s are set to be $\omega^{(1)} < 1$ and $\omega^{(2)} = \omega^{(3)} = \omega^{(4)} = 1$. In the following numerical experiment we use the four stored patterns of 100-dimensional binary vectors that have been used in the previous study on the associative chaotic neural network of 100 neurons [2][3][4][11][12][14].

3 Retrieval Characteristics with the Weighted Pattern Storage

Figure 1 shows the time course of distances $d_p(t)$ between the output pattern of the associative chaotic neural network and the four stored patterns when the weights for the stored patterns are set to $\omega^{(3)} = 0.9$ and $\omega^{(1)} = \omega^{(2)} = \omega^{(4)} = 1$ so that the 3rd stored pattern (i.e., Pattern (c)) is stored weaker than the other stored patterns. The distance between the output and the pth stored pattern is defined by

$$d_p(t) = \sum_{i=1}^{100}\left|x_i(t) - x_i^{(p)}\right|$$

where the pth patterns for $p = 1, 2, 3, 4$ correspond the patterns (a)-(d) in each row of Fig.1 and the patterns in Table 1, respectively. When the network retrieves the pth stored pattern and its reverse one exactly, d_p becomes 0 and 100, respectively. From Fig.1, we find that the network with the weighted auto-associative matrix retrieves the weaken stored pattern less frequent than the other stored patterns. It must be noted that the associative chaotic neural network with the conventional auto-associative network retrieves all the stored patterns in almost equal frequency[4]. Figure 1 also shows that the dynamics of the network looks non-periodic in this case. Spatiotemporal pattern representation of the output of the network in the same condition with Fig.1 is shown in Fig.2.

In order to see the effect of the weighted auto-associative matrix on the retrieval characteristics of the network in systematic way, retrieval frequencies of the patterns in the simulation during 10000 iterations (i.e., $t = 10001 \sim 20000$) are counted for each run with many conditions. The results are summarized in

Table 1: Retrieval frequencies of stored patterns with the weighted auto-associative matrix, where parameters are the same with the simulation of Fig.1 except for the weights for the stored patterns ($\omega^{(p)}$'s). The frequencies are counted during iterations of $t = 10001 \sim 20000$ for each row in the table.

Weaken Pattern	Init. Pat.	Pattern(a)		Pattern(b)		Pattern(c)		Pattern(d)	
		Exact	Reverse	Exact	Reverse	Exact	Reverse	Exact	Reverse
(a) ($\omega^{(1)} = 0.9$)	(a)	0	10	1	75	4	115	2	67
	(b)	0	9	0	54	0	91	5	83
	(c)	0	6	1	80	5	101	3	82
	(d)	1	8	1	73	1	103	2	80
(b) ($\omega^{(2)} = 0.9$)	(a)	5	108	0	2	1	85	9	161
	(b)	6	115	0	6	3	91	2	143
	(c)	2	104	0	4	2	110	4	122
	(d)	10	147	0	1	1	80	2	125
(c) ($\omega^{(3)} = 0.9$)	(a)	5	136	2	99	0	6	2	101
	(b)	3	127	2	93	0	5	5	103
	(c)	7	150	1	95	0	2	4	130
	(d)	6	129	3	93	0	4	2	124
(d) ($\omega^{(4)} = 0.9$)	(a)	2	87	3	89	2	72	0	3
	(b)	1	102	2	90	1	57	0	11
	(c)	4	104	3	105	2	84	0	8
	(d)	1	69	2	87	2	81	0	8

Table 1. It is shown in Table 1 that the retrievals of the weaken pattern with the weighted auto-associative matrix is less frequent than the other stored patterns in all the cases in the table.

4 Conclusions and Discussions

The associative chaotic neural network with the weighted auto-associative matrix retrieves the weaken stored pattern less frequent than the other stored patterns. By virtue of the weighted auto-associative matrix, one can adjust the strength of memory for each stored pattern and the strength may be represented as the retrieval frequency of the stored pattern in the associative chaotic neural network.

It is a future problem to study the relationship between the weights of the stored patterns and the dynamical properties of the network.

References

[1] K. Aihara, T. Takabe & M. Toyoda, "Chaotic Neural Networks," *Phys. Lett. A*, 144, 333–340, 1990.

[2] M. Toyoda, K. Aihara, K. Shimizu, M. Adachi & M. Kotani, "Chaotic Neural Networks," *Proc. of SICE'89*, 1323–1325, 1989.

[3] K. Aihara, "Chaotic Neural Networks," in *Bifurcation Phenomena in Nonlinear Systems and Theory of Dynamical Systems*, ed. H. Kawakami, World Scientific, Singapore, 1990, 143–161.

[4] M. Adachi & K. Aihara, "Associative Dynamics in a Chaotic Neural Network", *Neural Networks*, 10, 83–98, 1997.

[5] J. A. Anderson, "A simple neural network generating interactive memory," *Math . Biosciences*, 14, 197–220, 1972.

[6] T. Kohonen, "Correlation matrix memories," *IEEE Trans.*, C-21, 353–359, 1972.

[7] K. Nakano, "Associatron — a model of associative memory," *IEEE Trans.*, SMC-2, 381–388, 1972.

[8] H. Wigström, "A neuron model with learning capability and its relation to mechanism of association," *Kybernetic*, 12, 204–215, 1973.

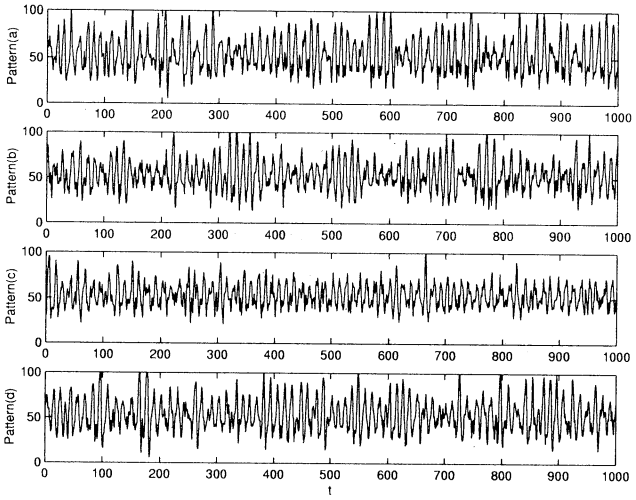


Figure 1: Distances $d_p(t)$ between the output pattern of the network and the four stored patterns with the weighted auto-associative matrix, where the parameters are set to $\omega^{(3)} = 0.9$ and $\omega^{(1)} = \omega^{(2)} = \omega^{(4)} = 1$; the other parameters are set to the standard values in the previous studies as follows: $\varepsilon = 0.015, k_f = 0.2, k_r = 0.9, \alpha = 10$ and $a_i = 2$ for all i.

[9] J.J. Hopfield, "Neural networks and physical systems with emergent collective computation abilities," *Proc. of Natl. Acad. of Sci.*, USA, 79, 2445–2558, 1982.

[10] Y. Inuma, T. Yoshinaga & H. Kawakami, "Bifurcation and chaotic itinerancy in a chaotic neural networks, part II," in *Proc. NOLTA'98*, 699–702, 1998.

[11] M. Adachi & K. Aihara, "Associative Chaotic Neural Networks with Asynchronous Updating: Winner updates faster," in *Proc. ICONIP'97*, 190–193, 1997.

[12] M. Adachi & K. Aihara, "Influence of Updating Rules on Retrievals in Associative Chaotic Neural Networks," in *Proc. NOLTA'97*, 645–648, 1997.

[13] S. Amari, "Characteristics of sparsely encoded associative memory," *Neural Networks*, 2, 451–457, 1989.

[14] M. Adachi & K. Aihara, "Complex dynamics of chaotic neural networks with an asynchronous updating" in *Proc. AROB'98*, 332-335, 1998.

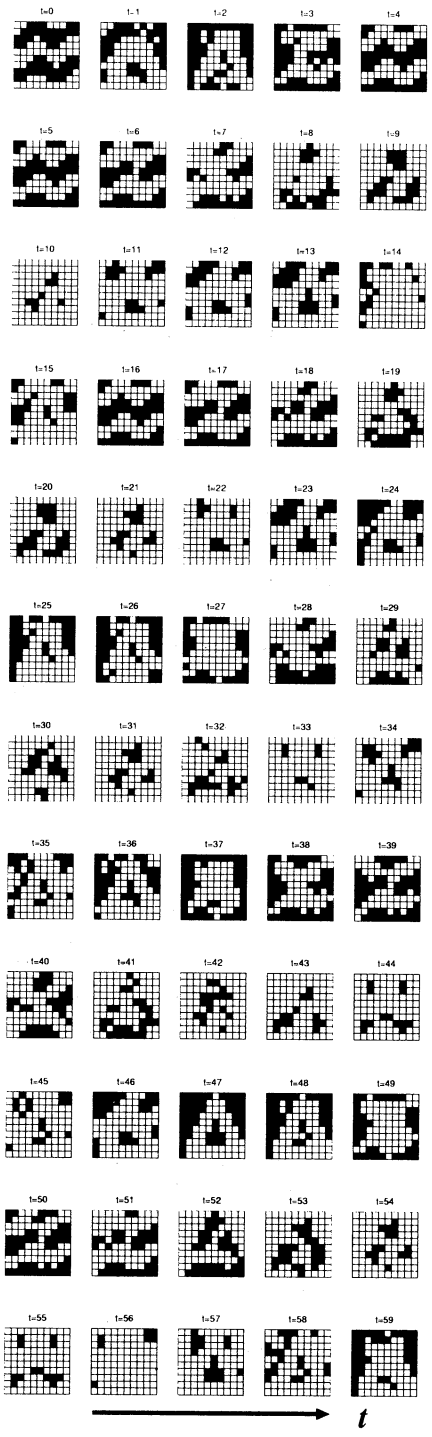


Figure 2: Spatiotemporal output patterns of the associative chaotic neural network in the same simulation of Fig.1. Where the pattern at each discrete time is displayed in the form of 10 by 10 matrices; the black and white squares represent 1 and 0, respectively.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Chaotic Evolution in a Game of a Host and Two Parasites

GEN HORI
Laboratory for Open Information Systems
Brain Science Institute, RIKEN
2-1, Hirosawa, Wako-shi, Saitama 351-0198

KAZU AIHARA
Department of Mathematical Engineering
The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo 113-8656

Abstract

A two-player game of a host and a parasite introduced by Ikegami and Yoshikawa[1] is extended to a game of a host and two parasites. Chaotic evolution of mixed strategies is numerically observed.

Keywords

game theory, mixed strategy, replication dynamics, chaos, bifurcation, Lyapunov exponent

1 Introduction

A two-player game is a fundamental model in the game theory and is defined by a gain structure matrix associated with pure strategy sets. Ikegami and Yoshikawa[1] introduced a two-player game of a host and a parasite, in which the parasite always exploit the host but occasionally both players gain positive returns, and a replication dynamics on the set of mixed strategies. The dynamical state of the evolution can be controlled through the mutation rates and symbiotic states of the host and the parasite appear in the chaotic region. In this article, we extend the game to the case of a host and two parasites and consider the dynamical state of the evolution of the game. Section 2 describes the game model introduced by [1], Section 3 introduces our extended model, and the final section contains remarks for future study.

2 Game of a host and a parasite

2.1 Game model

Ikegami and Yoshikawa[1] considered a game of two players, the host H and the parasite P. The host H and the parasite P have their own pure strategy sets $\{\alpha_0, \ldots, \alpha_{n-1}\}$ and $\{\beta_0, \ldots, \beta_{n-1}\}$, and payoff matrices A^H and A^P,

$$A^H = \begin{pmatrix} a^H_{0\,0} & \cdots & a^H_{0\,n-1} \\ \vdots & \ddots & \vdots \\ a^H_{n-1\,0} & \cdots & a^H_{n-1\,n-1} \end{pmatrix},$$

$$A^P = \begin{pmatrix} a^P_{0\,0} & \cdots & a^P_{0\,n-1} \\ \vdots & \ddots & \vdots \\ a^P_{n-1\,0} & \cdots & a^P_{n-1\,n-1} \end{pmatrix}.$$

When the host H and the parasite P take the pure strategies α_i and β_j respectively, they gain a^H_{ij} and a^P_{ij} respectively. The parasite's gain a^P_{ij} is given as a random number uniformly distributed on the interval [0,1),

$$a^P_{ij} \sim U[0,1).$$

On the other hand, the host's gain a^H_{ij} is basically given as

$$a^H_{ij} = -a^P_{ij},$$

but for a small number of the pairs (i,j) which are randomly selected, it is altered as

$$a^H_{ij} = -a^P_{ij} + \Delta$$

where Δ is a constant positive value.

The mixed strategies of H and P,

$$h = (h_0, \ldots, h_{n-1}), \quad h_i \geq 0, \quad \sum_{i=0}^{n-1} h_i = 1,$$

$$p = (p_0, \ldots, p_{n-1}), \quad p_j \geq 0, \quad \sum_{j=0}^{n-1} p_j = 1,$$

are the probability distributions on the pure strategy sets $\{\alpha_0, \ldots, \alpha_{n-1}\}$ and $\{\beta_0, \ldots, \beta_{n-1}\}$. The set of all the mixed strategies of each player forms a $(n-1)$-simplex. When H and P take the mixed strategies h and p, their expected returns are given as

$$hA^H p^t = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} a^H_{ij} h_i p_j,$$

$$hA^P p^t = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} a_{ij}^P h_i p_j.$$

They introduced population dynamics on the sets of mixed strategies by taking each pure strategy as a genotype, each mixed strategy as an ensemble and expected returns as fitness functions. Under the assumption that H alters h to maximize his expected return $hA^H p^t$ and P does p to maximize $hA^P p^t$, the reproduction is defined as

$$h_i' = \frac{\sum_{k=0}^{n-1} a_{ik}^H p_k + c}{hA^H p^t + c} h_i,$$

$$p_j' = \frac{\sum_{k=0}^{n-1} a_{kj}^P h_k + c}{hA^P p^t + c} p_j,$$

where $c > 1$ is a constant positive factor for preventing the system from diverging and keeping h_i's and p_j's positive. The denominators are for normalization of each mixed strategy and the $(n-1)$-simplex is invariant under the dynamics.

To introduce the mutation, n is assumed to be the Lth power of 2 for some integer L, and each pure strategy is represented by a binary sequence of length L. Each pure strategy α_k or β_k can mutate into its 1-bit neighborhood $N(k)$. For example, in the case of $L = 4$, the 1-bit neighborhood of $11 = 1011_{(2)}$ consists of $10 = 1010_{(2)}$, $9 = 1001_{(2)}$, $15 = 1111_{(2)}$ and $3 = 0011_{(2)}$, that is, $N(11) = \{10, 9, 15, 3\}$. Then α_{11} can mutate to α_{10}, α_9, α_{15} or α_3. The mutation is defined as

$$h_i'' = (1-\mu^H)h_i' + \frac{\mu^H}{L}\sum_{k\subset N(i)} h_k',$$

$$p_j'' = (1-\mu^P)p_j' + \frac{\mu^P}{L}\sum_{k\subset N(j)} p_k',$$

where μ^H and μ^P are the mutation rates for H and P respectively.

The mapping of mixed strategies from (h,p) to (h'',p'') by the reproduction and the mutation defined above is each time step of the discrete dynamics on the set of mixed strategies.

2.2 Numerical experiments

Ikegami and Yoshikawa[1] numerically observed the evolution of the above mentioned game model for the case of $L = 7$(that is, $n = 128$), $\Delta = 2$ for 5% of whole pairs (i,j), and $c = 2$, from random initial mixed strategies. They fixed the mutation rate of the host μ^H and varied the mutation rate of the parasite μ^P from large to small, and observed that the evolution goes to a fixed point dynamics if μ^P is large, a periodic oscillation if μ^P is small, and a chaotic oscillation if μ^P is very small. They pointed out that the optimal average scores of both players seem to be attained at the edge of chaos and periodic windows. See [1] for detail.

3 Game of a host and two parasites

3.1 Game model

In this section, we extend the game of two players described in the previous section to the game of three players, the host H and the two parasites P and Q. The three players H, P and Q have their own pure strategy sets $\{\alpha_0,\ldots,\alpha_{n-1}\}$, $\{\beta_0,\ldots,\beta_{n-1}\}$ and $\{\gamma_0,\ldots,\gamma_{n-1}\}$. To describe the situation that the two parasites P and Q are exploiting the host H, we define two pay off matrices A^{HP} and A^{HQ} for H,

$$A^{HP} = \begin{pmatrix} a_{0\,0}^{HP} & \cdots & a_{0\,n-1}^{HP} \\ \vdots & \ddots & \vdots \\ a_{n-1\,0}^{HP} & \cdots & a_{n-1\,n-1}^{HP} \end{pmatrix},$$

$$A^{HQ} = \begin{pmatrix} a_{0\,0}^{HQ} & \cdots & a_{0\,n-1}^{HQ} \\ \vdots & \ddots & \vdots \\ a_{n-1\,0}^{HQ} & \cdots & a_{n-1\,n-1}^{HQ} \end{pmatrix},$$

a pay off matrix A^{PH} for P,

$$A^{PH} = \begin{pmatrix} a_{0\,0}^{PH} & \cdots & a_{0\,n-1}^{PH} \\ \vdots & \ddots & \vdots \\ a_{n-1\,0}^{PH} & \cdots & a_{n-1\,n-1}^{PH} \end{pmatrix},$$

and a pay off matrix A^{QH} for Q,

$$A^{QH} = \begin{pmatrix} a_{0\,0}^{QH} & \cdots & a_{0\,n-1}^{QH} \\ \vdots & \ddots & \vdots \\ a_{n-1\,0}^{QH} & \cdots & a_{n-1\,n-1}^{QH} \end{pmatrix}.$$

When the host H and the parasites P and Q take the pure strategies α_i, β_j and γ_k respectively, the host H gains a_{ij}^{HP} from P and a_{ik}^{HQ} from Q, and the parasites P and Q gain a_{ij}^{PH} and a_{ik}^{QH} from H respectively. The parasite's gains a_{ij}^{PH} and a_{ik}^{QH} are given as uniformly distributed random numbers,

$$a_{ij}^{PH} \sim U[0,1),\ a_{ik}^{QH} \sim U[0,1),$$

and the host's gains are basically given as

$$a_{ij}^{HP} = -a_{ij}^{PH},\ a_{ik}^{HQ} = -a_{ik}^{QH},$$

with the exception for a small number of the pairs (i,j) which are randomly selected,

$$a_{ij}^{HP} = -a_{ij}^{PH} + \Delta,\ a_{ik}^{HQ} = -a_{ik}^{QH} + \Delta,$$

where Δ is a constant positive value.

When H, P and Q take the mixed strategies,

$$h = (h_0, \ldots, h_{n-1}),\quad h_i \geq 0,\quad \sum_{i=0}^{n-1} h_i = 1,$$

$$p = (p_0, \ldots, p_{n-1}),\quad p_j \geq 0,\quad \sum_{j=0}^{n-1} p_j = 1,$$

$$q = (q_0, \ldots, q_{n-1}),\quad q_k \geq 0,\quad \sum_{k=0}^{n-1} q_k = 1,$$

their expected returns are calculated as

$$hA^{HP}p^t + hA^{HQ}q^t = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} a_{ij}^{HP} h_i p_j + \sum_{i=0}^{n-1}\sum_{k=0}^{n-1} a_{ik}^{HQ} h_i q_k,$$

$$hA^{PH}p^t = \sum_{i=0}^{n-1}\sum_{j=0}^{n-1} a_{ij}^{PH} h_i p_j,$$

$$hA^{QH}q^t = \sum_{i=0}^{n-1}\sum_{k=0}^{n-1} a_{ik}^{QH} h_i q_k.$$

Assuming that each player alters his mixed strategy to maximize his own expected return, the reproduction is defined as

$$h_i' = \frac{\sum_{l=0}^{n-1} a_{il}^{HP} p_l + c}{hA^{HP}p^t + c} \frac{\sum_{l=0}^{n-1} a_{il}^{HQ} q_l + c}{hA^{HQ}q^t + c} h_i,$$

$$p_j' = \frac{\sum_{l=0}^{n-1} a_{lj}^{PH} h_l + c}{hA^{PH}p^t + c} p_j,$$

$$q_k' = \frac{\sum_{l=0}^{n-1} a_{lk}^{QH} h_l + c}{hA^{QH}q^t + c} q_k,$$

where $c > 1$ is a constant positive factor. The mutation into the 1-bit neighborhood is defined as the same as the previous section,

$$h_i'' = (1-\mu^H)h_i' + \frac{\mu^H}{L}\sum_{l \subset N(i)} h_l',$$

$$p_j'' = (1-\mu^P)p_j' + \frac{\mu^P}{L}\sum_{l \subset N(j)} p_l',$$

$$q_k'' = (1-\mu^Q)q_k' + \frac{\mu^Q}{L}\sum_{l \subset N(k)} q_l',$$

where μ^H, μ^P and μ^Q are the mutation rates for H, P and Q respectively.

3.2 Numerical experiments

We observe the evolution of the above mentioned extended game model for the case of $L = 4$(that is, $n = 16$), $\Delta = 2$ for 5% of whole pairs (i,j), and $c = 2$, from random initial mixed strategies. We fix the mutation rate of the host H as $\mu^H = 0.05$, and vary the mutation rates of the parasites P and Q as $\mu^P, \mu^Q = 10^{-i}$ $(i = 1,2,3,4,5,6)$ severally.

Fig.1. shows the maximum Lyapunov exponent of the dynamics as a function of the mutation rates of the parasites μ^P and μ^Q and summarize the relation

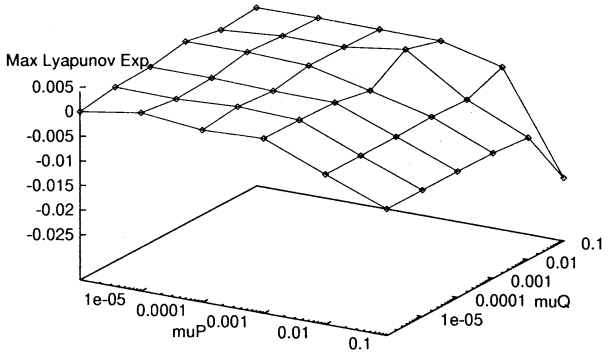


Fig.1.

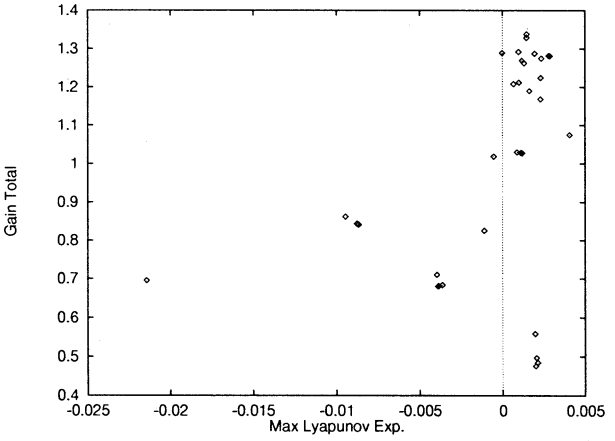


Fig.2.

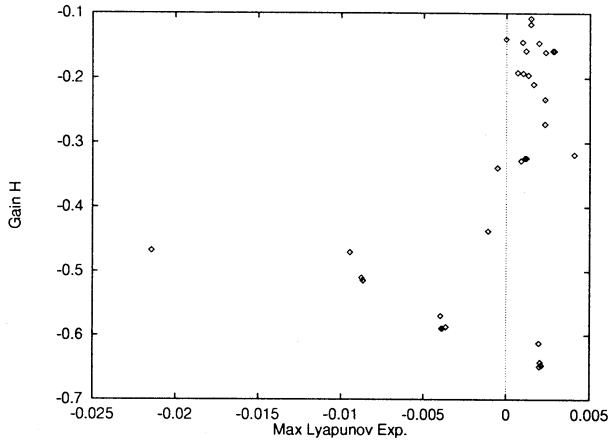


Fig.3.

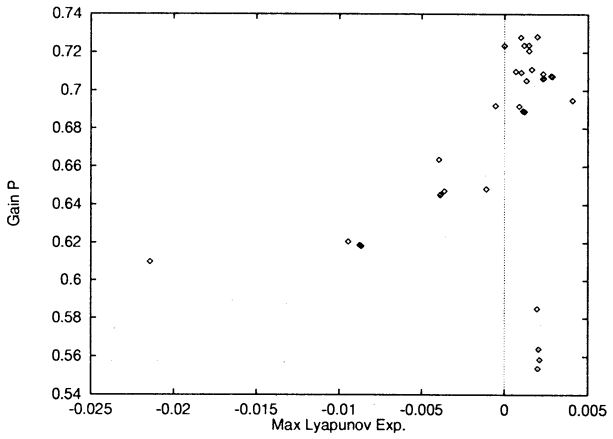


Fig.4.

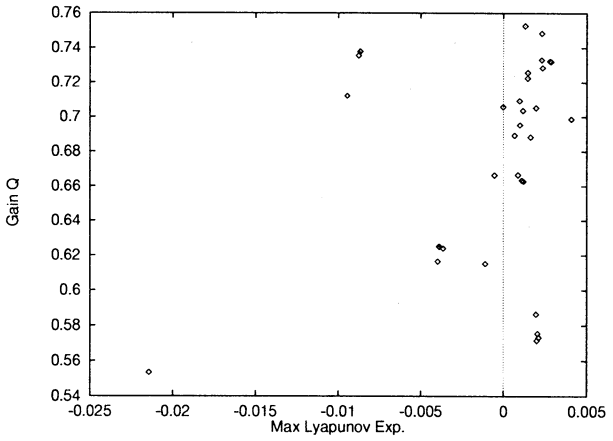


Fig.5.

between the dynamical state of the evolution and the mutation rates. From this, we see that both mutation rates μ^P and μ^Q have negative correlation to the maximum Lyapunov exponent. Fig.2-5. show the relation between the maximum Lyapunov exponent and the gains of the players. These show that the optimal states of the players are attained at the region where the maximum Lyapunov exponent is slightly larger than zero.

4 Remarks

We assume that there is no direct interaction between the parasites P and Q, in other words, $A^{PQ} = A^{QP} = 0$ in this article. The model with the direct interaction between P and Q, for example, some exclusive effect between P and Q, may be of interest. Also the exclusive effect between P and Q can be implemented in the framework of this article by putting Δ's of A^{HP} and Δ's of A^{HQ} on their exclusive columns. By this way, exclusiveness between P and Q can be gradually controlled. The numerical experiments for this case are left for future study.

There are other possibilities of extensions of games of three players, for example, i) C exploits B and B exploits A, ii) C exploits B, B exploits A and A exploits B etc. Also our framework can be extended to games of many players.

Acknowledgments

This work was partially done through the period in which one of the authors was a postdoctoral researcher of Japan Society for the Promotion of Science.

References

[1] T.Ikegami and E.S.Yoshikawa, "Chaos and Evolution of Cooperative Behavior in a Host-parasite Game", in *Towards the Harnessing of Chaos* (ed. M.Yamaguchi), Springer, pp.63-72, 1995.

[2] K.Kaneko and T.Ikegami, "Homeochaos: dynamic stability of a symbiotic network with population dynamics and evolving mutation rates", *Physica D*, 56, pp.406-429, 1992.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

IC Implementation of a Multi-Internal-State Chaotic Neuron Model with Unipolar and Bipolar Output Functions

Yoshihiko Horio, Izumi Kobayashi
Hiroshi Hayashi
Dept. of Electronic Eng., Tokyo Denki University
2-2, Kanda-Nishiki-cho, Chiyoda-ku
Tokyo, 101-8457, Japan

Kazuyuki Aihara
The Graduate School of Eng., The University of Tokyo
7-3-1, Hongo, Bunkyo-ku, Tokyo, 113-8656, Japan
and CREST, Japan Science & Tech., Co. (JST)
Kawaguchi, 332-0012, Japan

Abstract

A switched-capacitor circuit for the multi-internal-state chaotic neuron model is proposed. The proposed circuit has both of the bipolar and unipolar output functions. One of the two output functions can be chosen by external control voltage. The behavior of the circuit is confirmed using HSPICE simulations. Moreover, the proposed circuit is fabricated through MOSIS using 1.2 μm CMOS technology.
Keywords: Chaotic Neural Networks, Switched-Capacitor Circuits, Analog VLSI.

1 Introduction

Ever since the three-internal-state chaotic neuron model was proposed [1, 2], a multi-internal-state chaotic neural network has been applied to solve many problems such as combinatorial optimization problems [3]-[11] dynamical retrieval of memory states [12, 13], and so on. For example, the multi-internal-state is essential for solving some classes of optimization problems. The very high ability of the multi-internal-state chaotic neural network has been successively demonstrated in solving the quadratic assignment problems (QAPs) and the large-scale traveling salesman problems (TSPs) [7]-[11]. Moreover, the chaotic itinerancy has been observed in the dynamical associative memory (DAM) network with multi-internal-state chaotic neurons [12]. The chaotic itinerancy is an important system level behavior for dynamical information processing using chaos.

We have integrated switched-capacitor (SC) [6, 13]-[15] and switched-current (SI) chaotic neurons [5, 6, 16, 17, 18]. Furthermore, chaotic neural networks have been implemented using the integrated neurons, and have been applied to the DAM and TSP. However, all of the integrated neurons had single internal state. Therefore, the applications of the resulting neuron ICs are limited.

Furthermore, the integrated chaotic neurons had bipolar, that is, {-1,1}, output functions. The bipolar output function is suitable for some applications such as DAM network, while some others require unipolar, i.e., {0,1}, output functions. In ordinary neural networks, it is possible to convert or scale the problem itself from bipolar to unipolar and vice versa, so that neurons with either output function can be used. However, the variable transformation or scaling is not precisely applicable to the chaotic neuron model because of its nonlinear internal and external feedback paths.

In this paper, we propose a switched-capacitor circuit implementation of the multi-internal-state chaotic neuron model. The proposed circuit also has both of the bipolar and unipolar output functions. One of the two output functions can be chosen by external control voltage. The behavior of the circuit is confirmed using HSPICE simulations. The proposed circuit is currently under fabrication through MOSIS. The results from the chip will be presented at the conference.

2 Multi-Internal-State Chaotic Neuron Model

The multi-internal-state chaotic neuron model has been proposed in [2]. The dynamics of the i th chaotic neuron in a network with three internal states is described as

$$\begin{aligned} x_i(t+1) &= f\Bigg(\sum_{j-1}^{m} v_{ij} \sum_{d=0}^{t} k_e^d A_j(t-d) \\ &+ \sum_{j-1}^{n} w_{ij} \sum_{d=0}^{t} k_f^d x_j(t-d) \\ &- \alpha \sum_{d=0}^{t} k_r^d x_i(t-d) - \Theta_i \Bigg), \end{aligned} \tag{1}$$

where $x_i(t+1)$ is the output of i th chaotic neuron at discrete time $(t+1)$, m is the total number of external inputs to i th neuron, v_{ij} is the connection weight from j th external input, $A_j(t)$ is the strength of j th external input, n is the total number of neurons composing

the network, w_{ij} is the connection weight between j th and i th neurons, α is the parameter of refractoriness, k_e, k_f and k_r are the decay parameters for external inputs, feedback inputs from the other neurons in the network and self-refractoriness, respectively, Θ_i is the threshold of i the neuron, and $f(\cdot)$ is the nonlinear output function of the neuron.

Three internal states, that is, the internal state related to the external inputs, ξ_i, that for the feedback inputs from other neurons in the network, η_i, and the relative refractoriness, ζ_i, are defined as

$$\xi_i(t+1) = \sum_{j-1}^{m} v_{ij} \sum_{d=0}^{t} k_e^d A_j(t-d), \quad (2)$$

$$\eta_i(t+1) = \sum_{j-1}^{n} w_{ij} \sum_{d=0}^{t} f_e^d x_j(t-d), \quad (3)$$

$$\zeta_i(t+1) = -\alpha \sum_{d=0}^{t} k_r^d x_i(t-d) - \Theta_i, \quad (4)$$

respectively. The reduced equations for each internal state can be written by

$$\xi_i(t+1) = k_e \xi_i(t) + \sum_{j=0}^{m} v_{ij} A_j(t), \quad (5)$$

$$\eta_i(t+1) = k_f \eta_i(t) + \sum_{j=0}^{n} w_{ij} x_j(t), \quad (6)$$

$$\zeta_i(t+1) = k_r \zeta_i(t) - \alpha x_j(t) - \theta_i, \quad (7)$$

where θ_i is $\Theta_i(1-k_r)$. Furthermore, the total internal state of i th neuron, $y_i(t+1)$, is given by

$$y_i(t+1) = \xi_i(t+1) + \eta_i(t+1) + \zeta_i(t+1). \quad (8)$$

Finally, the output of i th neuron defined by eq. (1) can be written as

$$x_i(t+1) = f(\xi_i(t+1) + \eta_i(t+1) + \zeta_i(t+1)). \quad (9)$$

3 Realization of the Unipolar and Bipolar Output Functions

The straightforward way to realize a neuron circuit with both unipolar and bipolar output functions is to implement each output functions separately. However, this will cost large chip area, and require additional circuits to control the shape of each output function.

In this paper, a novel technique to obtain both output functions with only one nonlinear function circuit is proposed. In stead of having two nonlinear function circuits, two simple output buffer circuits are used.

A schematic for obtaining both of the bipolar and unipolar output functions with one nonlinear circuit is shown in Fig. 1. As shown in the figure, the bipolar and the unipolar output functions are denoted as $-f_B(\cdot)$ and $-f_U(\cdot)$, respectively. Note that the negative signs are used for these output functions in order to be consistent with the SC circuit implementation mentioned in sec. 4.

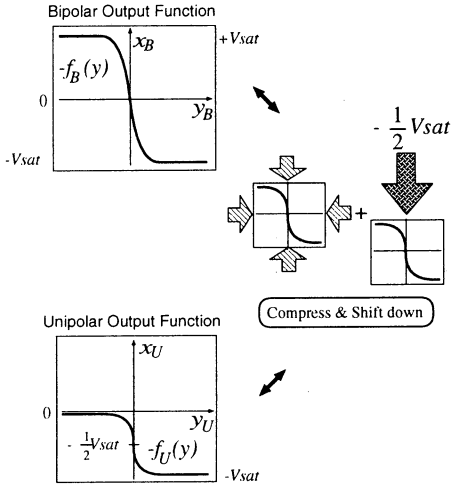


Figure 1: The transformation between two types of the output functions. $-f_U$ and $-f_B$ are used in the figure in order to be consistent with the SC circuit implementation.

In the following, the conversion scheme from the bipolar to unipolar is described as an illustration. The operations are two fold. First, the area of the bipolar output function is compressed by quarter (both of horizontal and vertical axes are compressed by half). Second, the bias level is shifted by $-\frac{1}{2}V_{sat}$, where V_{sat} is a saturation voltage of $-f_B(\cdot)$.

If we denote the horizontal and vertical axes of the bipolar function as y_B and x_B, respectively, while those for the unipolar one as y_U and x_U, respectively, the above operations can be written by

$$\begin{cases} x_U = \frac{1}{2}x_B - \frac{1}{2}V_{sat} = \frac{1}{2}(x_B - V_{sat}), \\ y_U = \frac{1}{2}y_B. \end{cases} \quad (10)$$

Through the above transformation, the similarity of the two functions are guaranteed. However, in the practical circuit implementation, lower equation in eq. (10) can not be realized while upper one can be. Therefore, we should keep the original horizontal axis. That is, we use

$$y_U = y_B, \quad (11)$$

instead of lower equation in eq. (10). In this case, the similarity between $f_U(\cdot)$ and $f_B(\cdot)$ does not hold anymore. Therefore, an approximation method is used to obtain the quasi-similarity. Namely, we only guarantee that the first derivatives of the two output functions at $y_U = y_B = 0$ are equal as

$$\left.\frac{\mathrm{d}f_U(y_U)}{\mathrm{d}y_U}\right|_{y_U=0} = \left.\frac{\mathrm{d}f_B(y_B)}{\mathrm{d}y_B}\right|_{y_B=0}. \qquad (12)$$

In other words, the gains of $f_U(\cdot)$ and $f_B(\cdot)$ at zero input are kept equal. As a result, it is shown that the center gain of nonlinear output function circuit should be adjustable according to the bipolar or unipolar output function characteristic.

4 Switched-Capacitor Multi-Internal-State Chaotic Neuron Circuit with Unipolar and Bipolar Output Functions

A schematic diagram of the switched-capacitor three-internal-state chaotic neuron circuit with unipolar and bipolar output functions is shown in Fig. 2. As shown in the figure, each internal state is realized by one SC damped-integrator circuit. The use of three separate operational amplifiers increases the chip area. However, it is easy to control the time constant for refractoriness of each internal state separately. Therefore, the circuit can be used in wide range of applications.

As shown in the figure, one OTA circuit is used to make a nonlinear characteristic. The center gain and saturation levels of the OTA should be adjustable as mentioned in sec. 3. Furthermore, two output stages, one is for bipolar and the other is for unipolar, are prepared, and one of them is selected externally. A special OTA circuit and simple output stages have been designed.

5 HSPICE Simulation Results

The bifurcation diagrams of the total internal state, $y(t_n)$, and the neuron output, $x(t_n)$, are shown in Fig. 3. In Figs. 3(a) and (b), results with unipolar output function are shown. Figures 3(c) and (d) shows results with bipolar output function.

6 Conclusions

A switched-capacitor multi-internal-state chaotic neuron circuit with unipolar and bipolar output functions has been proposed. The circuit is currently under

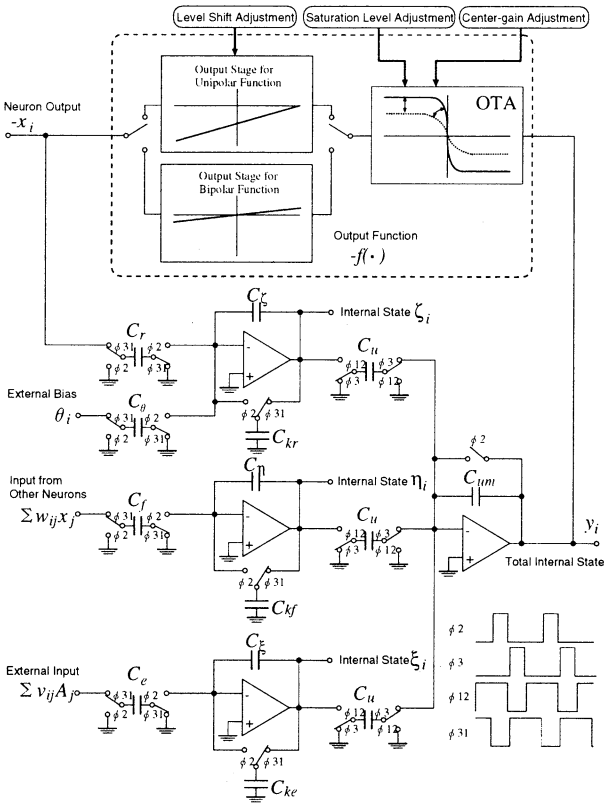


Figure 2: The schematic diagram of the switched-capacitor three-internal-state chaotic neuron circuit with unipolar and bipolar output functions.

fabrication through MOSIS using 1.2 μm CMOS technology. Results from the chip will be presented at the conference. Moreover, we are planning to construct a large chaotic neural network system with 10,000 neurons and 10^8 programmable synapses. The architecture of the system will be also shown at the conference.

References

[1] K. Aihara, T. Takabe, and M. Toyoda, 'Chaotic neural networks,' *Phys. Lett. A*, vol. 144, no. 6,7, pp. 333–340, 1990.

[2] K. Aihara, 'Chaotic neural networks,' in *Bifurcation Phenomena in Nonlinear System and Theory of Dynamical Systems*, H. Kawakami ed., pp. 143–161, World Scientific, 1990.

[3] L. Chen and K. Aihara, 'Chaotic simulated annealing by a neural network model with transient chaos,' *INNS Neural networks*, vol. 8, no. 6, pp. 915–930, 1995.

[4] Y. Horio, K. Suyama, A. Dec, and K. Aihara. 'Switched-capacitor chaotic neural networks

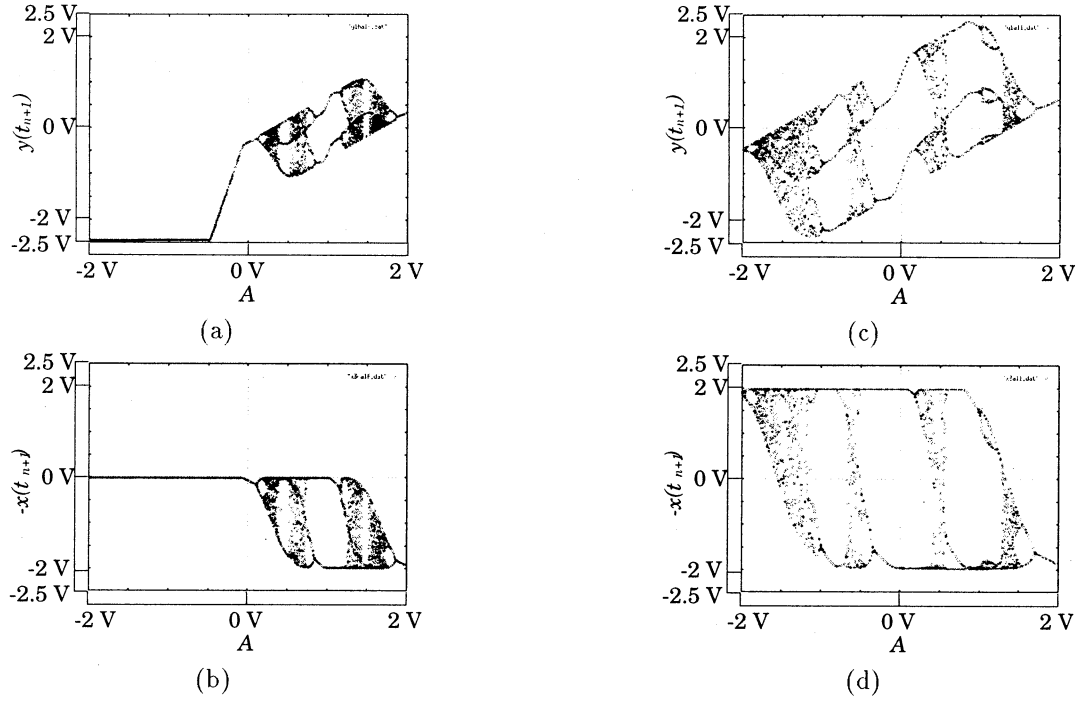


Figure 3: Simulation results of the SC chaotic neuron circuit. Bifurcation diagrams of (a) internal state and (b) output of the unipolar neuron, and (c) internal state and (d) output of the bipolar neuron.

for traveling salesman problem,' in *Proc. INNS WCNN'94*, vol. 4, pp. 690–696, 1994.

[5] N. Kanou and Y. Horio, 'Design of current-mode transiently chaotic neural networks for traveling salesman problem,' in *Proc. INNS WCNN'95*, vol. 1, pp. 282–285, 1995.

[6] Y. Horio, N. Kanou, and K. Suyama, 'Switched-capacitor and switched-current chaotic neural networks for combinatorial optimization problems,' in *Proc. NOLTA'95*, vol. 1 of 2, pp. 1–6, 1995.

[7] M. Hasegawa, T. Ikeguchi and K. Aihara, 'Solving traveling salesman problems using a heuristic method with chaotic neurodynamics,' in *Proc. NOLTA'96*, pp. 337–340, 1996.

[8] M. Hasegawa, T. Ikeguchi and K. Aihara, 'A novel chaotic search for combinatorial optimization,' in *Proc. NOLTA'97*, pp. 613–616, 1997.

[9] M. Hasegawa, and T. Ikeguchi, 'Combination of chaotic neurodynamics with the 2-opt algorithm to solve traveling salesman problems,' *Physical Review Lett.*, vol. 79, no. 12, pp. 2344–2347, 1997.

[10] M. Hasegawa, T. Ikeguchi and K. Aihara, 'Solving quadratic assignment problems by chaotic neural networks,' in *Proc. ICONIP'97*, pp. 182–185, 1997.

[11] M. Hasegawa, T. Ikeguchi and K. Aihara, 'A novel approach for solving large scale traveling salesman problems by chaotic neural networks,' in *Proc. NOLTA'98*, pp. 711–714, 1998.

[12] M. Adachi and K. Aihara, 'Associative dynamics in a chaotic neural network,' *INNS Neural networks*, vol. 10, no. 1, pp. 83–98, 1997.

[13] Y. Horio and K. Suyama, 'Dynamical associative memory using integrated switched-capacitor chaotic neurons,' in *Proc. IEEE ISCAS'95*, vol. 1, pp. 429–432, 1995.

[14] Y. Horio and K. Suyama, 'Switched-capacitor chaotic neuron for chaotic neural networks,' in *Proc. IEEE ISCAS'93*, pp. 1018–1021, 1993.

[15] Y. Horio and K. Suyama, 'IC implementation of switched-capacitor chaotic neuron,' in *Proc. IEEE ISCAS'94*, vol. 6 of 6, pp. 97–100, 1994.

[16] N. Kanou, Y. Horio, K. Aihara and S. Nakamura, 'A current-mode circuit of a chaotic neuron model,' *IEICE Trans. on Fundamentals*, vol. E76-A, no. 4, pp. 642–644, 1993.

[17] N. Kanou, Y. Horio, K. Aihara and S. Nakamura, 'A current-mode implementation of a chaotic neuron model using a SI integrator.' *IEICE Trans. on Fundamentals*, vol. E77-A, no. 1, pp. 335–338, 1994.

[18] R. Herrera, K. Suyama, and Y. Horio, 'IC implementation of a current-mode chaotic neuron,' in *Proc. IEEE ISCAS'98*, vol. 3, pp. 546–549, 1998.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Detecting Nonlinear Causality via Nonlinear Modeling

Tohru Ikeguchi
Department of Applied Electronics
Science University of Tokyo
2641 Yamazaki, Noda, Chiba 278–8510 Japan

Abstract

We analyze a set of complex time series from the view point of nonlinear causality. The mathematical background for analyzing time series is an extension of embedding theories of autonomous systems to an input–output system. We consider that the existence of nonlinear causality can be detected by nonlinear predictability of input and output sequences. Several numerical examples are given for confirmation of the proposed framework.

1 Introduction

Many researches on deterministic chaos have been made during the last two decades, then we can understand complex behavior in natural worlds from the view point of nonlinear low-dimensional dynamical systems and deterministic chaos. It has been clarified that low-dimensional deterministic nonlinear systems can naturally produce complicated chaotic behavior. Moreover, it has been reported that existence of low dimensional chaos have been discovered in many experimental fields.

As characteristics of deterministic chaos have been revealed, these characteristics have strongly affected on a conventional time series analysis. Time series analysis methods are reincarnated as a novel one which claims that even if an observed time series appears irregular, it might be produced from a low-dimensional nonlinear dynamical system, then its behavior might be chaotic. Then identifying deterministic chaos and its quantitative characterization are very important from the viewpoint of time series analysis on nonlinear dynamical systems theory. For quantitative characterization of deterministic chaos, there are several statistics, for example, the fractal dimensions, the Lyapunov exponents, the metric entropies and so on.

In order to analyze complex time series from the view point of nonlinear dynamical system theory, it is usually assumed that the time series has been produced from autonomous systems. Then, celebrated results from Takens' embedding theory [8] and its extension by Sauer et al. [5] give the mathematical foundation for recovering the nonlinear dynamical systems which are usually unknown. However, in the natural world, it is not so rare that we treat or measure the time series which might be thought as an output from a non-autonomous system, or an input-output system. However, it is sometimes impossible to obtain information of an input-output system, but we can only access to input and output sequences. In such a case, it might be desired to detect such kinds of input and output relations only from observed sequences.

In this paper, we consider the above situation, and raise an issue of possibility for detecting such kind of relation, which we shall call "nonlinear causality," even if we can only touch input and output time series.

2 Embedding theory for input–output systems

Let us first consider the relation of a dynamical system and an observation function for an input–output system.

$$\begin{cases} \boldsymbol{x}(t+1) &= \boldsymbol{f}_{\boldsymbol{\mu}}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \boldsymbol{u}(t+1) &= \boldsymbol{g}(\boldsymbol{u}(t)) \\ \boldsymbol{y}(t) &= \boldsymbol{h}(\boldsymbol{x}(t)) \end{cases} \tag{1}$$

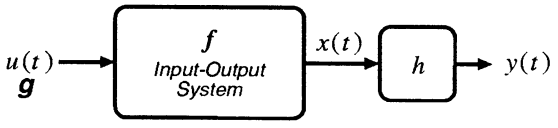


Figure 1: Conceptual representation of an input–output system. In this figure, $(\boldsymbol{f}, \boldsymbol{g}, \boldsymbol{h})$ is assumed to be unknown.

where $\boldsymbol{x}$ is a state vector, $\boldsymbol{f}$ is a dynamical system with a parameter vector $\boldsymbol{\mu}$, $\boldsymbol{u}$ is an input of the system, $\boldsymbol{g}$ is a dynamics for inputs, and $\boldsymbol{h}$ describes an observation function.

With the above situation, we will argue that the construction of a delay coordinate by the following form

$$\begin{aligned} \boldsymbol{v}(t) = \{&y(t), y(t-\tau), \ldots, y(t-(m-1)\tau), \\ &u(t), u(t-\tau), \ldots, u(t-(l-1)\tau)\} \end{aligned} \quad (2)$$

will be embedding, if the reconstructed dimensions m and l are sufficiently large. Recently, the Takens embedding theorem has been extended to the above kind of the input–output system by Stark [6, 7]. With the analogy of an extension of the original Takens' theorem to the fractal prevalent version by Sauer et al. [5], we will consider the following new embedding theorem, which might be hold for an input–output system [1].

Theorem 2.1 (Embedding Theorem) *Let $\boldsymbol{f}$ be a diffeomorphism in $\mathbf{R}^k$, A be an attractor of $\boldsymbol{f}$, D_0 be the box count dimension of A. Let assume that $m > 2D_0$ and $l > 2D_0$, that for all positive integers $p \leq m$, the box count dimension of the set A_p of periodic points with the period p is smaller than $p/2$, and that the Jacobian matrix Df^p for each of these periodic orbits has different eigenvalues.*

Then, for almost every smooth function h, the transformation of Eq.(3) is topological and differential embedding.

$$\begin{aligned} \boldsymbol{v}(t) = \{&h(\boldsymbol{f}(\boldsymbol{x})), h(\boldsymbol{f}^2(\boldsymbol{x})), \ldots, h(\boldsymbol{f}^{m-1}(\boldsymbol{x})), \\ &u(t), u(t-1), \ldots, u(t-(l-1))\} \end{aligned} \quad (3)$$

The above theorem describes that with sufficiently large reconstructed dimension, we can have at least one-to-one correspondence between the input–output dynamics $\boldsymbol{f}$ and $\boldsymbol{P}$ which is described by the following function,

$$\begin{aligned} \boldsymbol{v}(t+1) = \boldsymbol{P}(\{&y(t), y(t-1), \ldots, y(t-(m-1)), \\ &u(t), u(t-1), \ldots, u(t-(l-1))\}). \end{aligned} \quad (4)$$

Now, we define the existence of nonlinear causality by nonlinear predictabilities of an input–output system formulated above, since if the transformation vector of Eq.(3) could be embedding, Eq.(3) is a proper form of representing the reconstructed state space, therefore estimating a smooth function $\boldsymbol{P}$ in Eq.(4) could be utilized as a predictor for the state space of Eq.(3).

3 Producing Input–Output Sequences

In this paper, we utilize simple causal systems for making input and output sequences. The first example is the ARMA type:

$$\boldsymbol{f}(x) = \sum_i^M a_i x(t-i)^{p_i} + \sum_j^N b_j u(t-j)^{q_j} \quad (5)$$

where a_i and b_i are parameters, and p_i and q_j are order of nonlinearities. If we set $p_i = 1, q_j = 1, \forall i, j$, it represents the linear ARMA model.

The second example is a forced Ikeda map[3], which is described by the following equations:

$$\begin{cases} x_1(t+1) = q + b(x_1(t)\cos\theta - x_2(t)\sin\theta))) + u(t) \\ x_2(t+1) = \quad b(x_1(t)\sin\theta + x_2(t)\sin\theta))) \end{cases} \quad (6)$$

where $\theta = \kappa - \dfrac{\alpha}{1 + x_1(t)^2 + x_2(t)^2}$ and $u(t)$ is a forcing input. We set the parameter values of the Ikeda map to a typical set which produces a chaotic response ($q = 1.0, b = 0.7, \kappa = 0.4, \alpha = 6.0$). In this paper, we judge chaotic response by existence of a positive Lyapunov exponent.

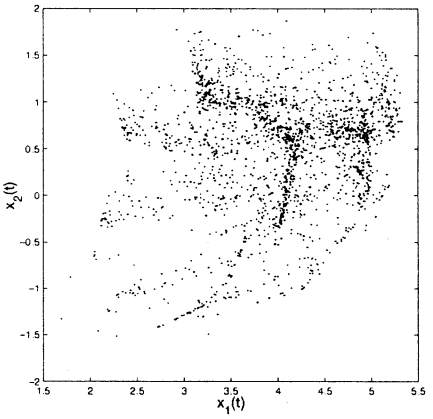


Figure 2: State space representation of a forced Ikeda map driven by a nonlinear deterministic system. The forced system is the Hénon map [2].

4 Predicting Input–Output sequences

As for the algorithm for predicting time series, we utilize a simple technique called the method of analogues [4], which is 0-th order approximation. For the

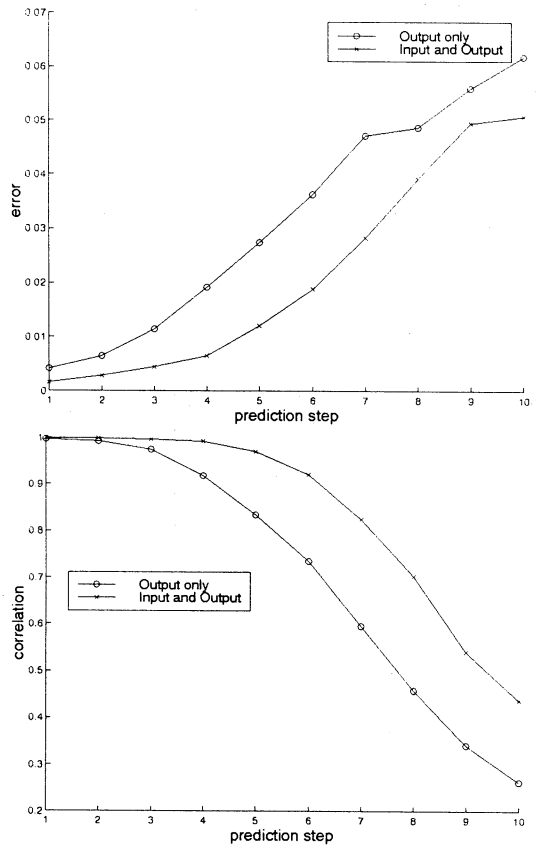


Figure 3: Nonlinear predictability of an input–output system defined by Eq.(5). Parameters are set to $a_i = 0, \forall i$, $b_1 = b_2 = 1, q_1 = q_2 = 1$ and $b_j = 0, \forall j (\neq 1, 2)$.

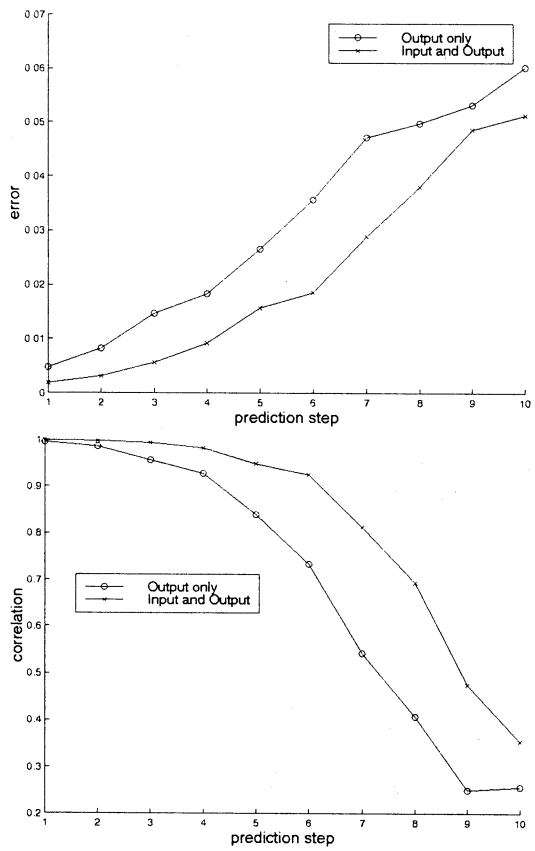


Figure 4: Nonlinear predictability of an input–output system defined by Eq.(5). Parameters are set to $a_i = 0, \forall i$, $b_1 = b_2 = 1, q_1 = 2, q_2 = 1$ and $b_j = 0, \forall j (\neq 1, 2)$.

evaluation of nonlinear predictability, we use two measures; the first is normalized prediction error and the second is correlation coefficients between the predicted time series and the actual time series.

In Fig.3, the results of a simple case are shown, where the input–output system $\boldsymbol{f}$ is linear MA. From Fig.3, when the input sequence $u(t)$ is involved for constructing a state space, prediction performance increase. The same tendencies are observed for a more complicated case shown in Fig.4; it is the case that the input–output system $\boldsymbol{f}$ is nonlinear MA. These results imply that it is possible to detect nonlinear causality by nonlinear modeling with constructing state space in the form of Eq.(2).

In Fig.5, we show the results from the second example. The Ikeda map is forced by an input sequence. In Fig.5, $u(t)$ is a deterministic nonlinear sequence, i.e. the first variable of the Hénon map. Since the box count dimension of the Ikeda map is estimated almost ~ 1.5, then we set the reconstructed dimensions as $m = 3$ and $l = 3$.

For comparison, we introduce the method of surrogate data [9] in order to check the augmented information by input sequences really affects the increase of prediction performance. Namely, we construct a state space for prediction by the following:

$$\boldsymbol{v}(t+1) = \{y(t), y(t-1), \ldots, y(t-(m-1)), \\ u'(t), u'(t-1), \ldots, u'(t-(l-1))\}. \quad (7)$$

where $u'(t)$ denotes a surrogate sequence of $u(t)$.

In Fig.5, we show the results of applying nonlinear prediction algorithm to the three cases; namely the state space is made (1) only from output sequences, (2) from both input and output sequences, and (3) from output and surrogate of input sequences.

From the results, we can see that the best prediction performance is obtained by the case of constructing

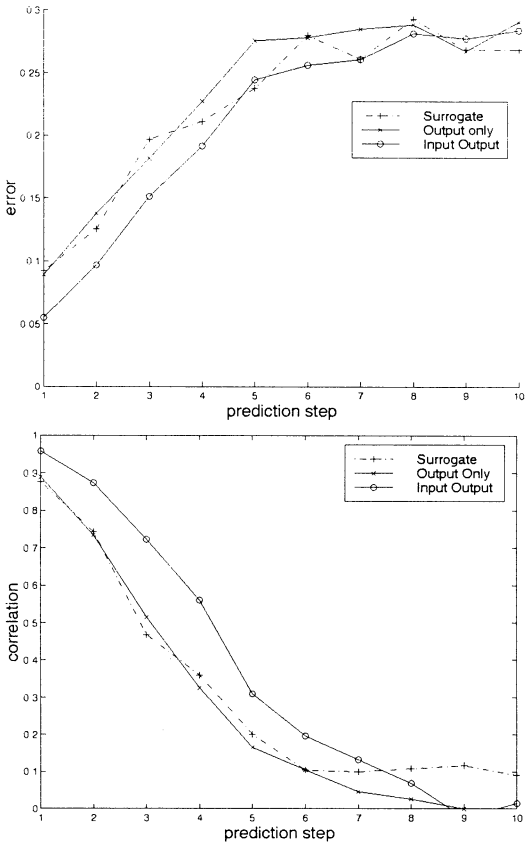


Figure 5: Prediction performance of input–output system prediction. The algorithm for making surrogate data is amplitude adjusted Fourier transform.

the state space with both input and output sequences. For the other reconstructions, prediction performance becomes worse. Moreover, for the case that the augmented information is made from the surrogate of $u(t)$, nonlinear predictability is almost the same as the case that the state space is only made with the output sequences, which means that information of input sequences is important for reconstructing unknown dynamics.

5 Conclusion

We analyze sets of complex time series by the nonlinear prediction scheme in order to detect nonlinear causality. We propose a method for detecting nonlinear causality by nonlinear prediction with the mathematical foundation of the embedding theorem for an input–output system. We apply the proposed algorithm to numerical examples from mathematical models of nonlinear dynamical systems. As a result, it is possible to detect nonlinear causality between input sequences and output sequences, since nonlinear predictabilities increase with the information of input sequences.

In this paper, we only give an intuitive notation of the proposed embedding theorem for an input–output system [1], a mathematical rigorous proof should be given.

This research was partly supported by Grant-in-Aid from the Ministry of Education, Culture and Science of Japan (No.10750287).

References

[1] M. Casdagli. "A Dynamical Systems Approach to Modeling Input–Output Systems". In M. Casdagli and S. Eubank, editors, *Nonlinear Modeling and Forecasting*, Vol. XII, pp. 265–281. Addison-Wesley, 1992.

[2] M. Hénon. "A Two-dimensional Mapping with a Strange Attractor". *Communications in Mathmatical Physics*, Vol. 50, pp. 69–77, 1976.

[3] K. Ikeda. "Multiple-Valued Stationary State and Its Instability of The Transmitted Light By A Ring Cavity System". *Optics Communications*, Vol. 30, No. 2, pp. 257–261, August 1979.

[4] E. N. Lorenz. "Atomospheric Predictbility as Revealed by Naturally Occuring Analogues". *Journal of the Atomospheric Sciences*, Vol. 26, pp. 636–646, July 1969.

[5] T. Sauer, J. A. Yorke, and M. Casdagli. "Embedology". *Journal of Statistical Physics*, Vol. 65, No. 3/4, pp. 579–616, 1991.

[6] J. Stark. "Delay Embeddings For Forced Systems: I. Deterministic Forcing", 1998. to appear in Journal of Nonlinear Science.

[7] J. Stark, D. S. Broomhead, M. E. Davies, and J. Huke. "Takens Embedding Theorems for Forced and Stochastic Systems". *Nonlinear Analysis*, Vol. 30, pp. 5303–5314, 1997.

[8] F. Takens. "Detecting strange attractors in turbulence". In D. A. Rand and B. S. Young, editors, *"Dynamical Systems of Turbulence"*, Vol. 898 of *Lecture Notes in Mathematics*, pp. 366–381, Berlin, 1981. Springer-Verlag.

[9] J. Theiler, S. Eubank, A. Longtin, B. Galdrikian, and J. D. Farmer. "Testing for nonlinearity in time series : the method of surrogate data". *Physica D*, Vol. 58, pp. 77–94, 1992.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Short-term prediction about complex sequences in a blast furnace by bell-shaped radial basis function networks

Takaya Miyano
Sumitomo Metal Industries, Ltd.
1-8 Fusocho, Amagasaki, Hyogo 660-0891, Japan

Hiroshi Shibuta
Sumitomo Metal Industries, Ltd.
1850 Minato, Wakayama 664, Japan

Kazuyuki Aihara
The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

Abstract

Nonlinear dynamical analysis is applied to time series data of temperature fluctuations actually observed in a blast furnace. The thermal sequence seems to be chaotic when no control actions are taken to the plant. Generalized radial basis function networks with Gaussian functions and $(1+\cosh x)^{-1}$ as basis functions are found to be capable of learning the underlying dynamics and of making short-term forecasts about the thermal sequence.

1 Introduction

Dynamical behaviors of actual systems and plants are observed in most situations as complicated sequences with irregularity. The discovery of chaos has shown that determinism can lurk in such random sequences, which may suggest effective control of actual systems by capturing nonlinear determinism underlying irregular behavior with nonlinear predictors such as neural networks. In this paper we examine dynamical properties of time series data of temperature fluctuations actually observed in a blast furnace as a huge chemical reactor for iron making, aiming at constructing a nonlinear predictor for forecasting irregular dynamical behavior of the thermal sequence.

2 Time Series Data

A blast furnace with a size of $\sim$100 m in height and $\sim$10 m in diameter is equipped with cooling apparatus for water–cooling its side wall to prevent from damage by heat. The time series data to be analyzed are the temperature difference between input and output cooling water. Various actions for plant control are usually taken by reference to the trends of the temperature sequences. Figure 1 shows a time series of the temperature difference for which no control actions were taken during measurement. The time series consists of 528 data points observed at a sampling time of 10 min, including no regularities. Previous work based on KM_2O Langevin formalism has shown that the time series is stationary in terms of weak stationarity [1].

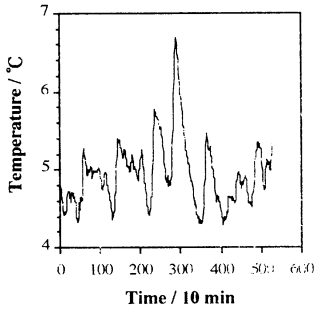


Figure 1. Time series of temperature differences between input and output cooling water observed for cooling apparatus in a blast furnace.

3 Time Series Analysis

A Sugihara-May algorithm [2] is applied to the time series data to examine the scaling property of the predictive error with respect to the prediction time interval. In the Sugihara–May method we first construct input vectors consisting of lagged sequences of data points as $\boldsymbol{x}(t) = \{x(t), x(t-\Delta t), \cdots, x(t-(D-1)\Delta t)\}$ and the corresponding output values as $x(t+\tau\Delta t)$ from a time series $\{x(t)\}$, where D is the embedding dimension, $\Delta t = 10$ min is the sampling time interval, and τ is the prediction time interval. Let pairs of $\boldsymbol{x}(t_k)$ and

$x(t_k + \tau\Delta t)$ generated from the first half of the time series be library patterns representing the dynamical behavior of the data. Then the output values to $\boldsymbol{x}(t_p)$ can be predicted by the following equation.

$$y(t_p + \tau\Delta t) = \frac{\sum_{k=1}^{D+1} x(t_k + \tau\Delta t)\exp(-d_k)}{\sum_{k=1}^{D+1}\exp(-d_k)} \quad (1)$$

$$d_k = |\,\boldsymbol{x}(t_p) - \boldsymbol{x}(t_k)\,| \quad (2)$$

where the summation is taken over the library patterns $\boldsymbol{x}(t_k)$ (k running from 1 to $D+1$) as $D+1$ closest neighbors forming the vertices of the smallest simplex including $\boldsymbol{x}(t_p)$ in the D–dimensional Euclidean space. Figure 2 shows the scaling property of the predictive error $E(\tau)$ as a function τ, where $E(\tau)$ denotes the root–mean–squared error between predicted values and the corresponding actual values normalized by the standard deviation of the actual data. The optimal embedding dimension corresponds to $D = 3$ in terms of $E(1)$ (the predictive error at $\tau = 1$) as a function of D. Linear correlation of the semilog plot of τ versus $\log[E(\tau)]$ is estimated to be 0.999 for each choice of D in Fig.2. This indicates an exponential decay of predictability with time as a signature of possible existence of deterministic chaos.

Another diagnostic test for determinism is based on

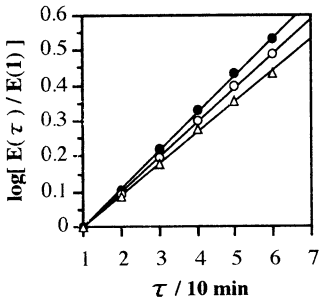


Figure 2. Semilog plot of the predictive error as a function of the prediction time interval for forecasts about the temperature sequence with a Sugihara-May predictor at $D = 3$ (●), 6 (○), 10 (△). Forecasts are made about the last half of the time series using the first half as library patterns.

the algorithm developed by Wayland et. al. [3]. We again make use of the embedding space consisting of $\boldsymbol{x}(t)$. The central idea of this method is that neighboring vectors should point in similar directions, if determinism is visible in the time series. Let $\boldsymbol{x}(t + T\Delta t)$ be the image of $\boldsymbol{x}(t)$, where T is an appropriately chosen time interval. We first find K nearest neighbors of $\boldsymbol{x}(t_0)$ in the sense of the Euclidean distance, denoted as $\boldsymbol{x}(t_k)$ ($k = 0, 1, \cdots, K$). We next calculate translation vectors as

$$\boldsymbol{v}(t_k) = \boldsymbol{x}(t_k + T\Delta t) - \boldsymbol{x}(t_k) \quad (3)$$

for $\boldsymbol{x}(t_k)$ and the corresponding images. The diversity of directions of nearby trajectories, in other words, degrees of visible determinism of the data can be gauged in terms of a useful measure referred to as translation error E_{trans}:

$$E_{trans} = \frac{1}{K+1}\sum_{k=0}^{K}\frac{|\,\boldsymbol{v}(t_k) - \hat{\boldsymbol{v}}\,|^2}{|\,\hat{\boldsymbol{v}}\,|^2}, \quad (4)$$

$$\hat{\boldsymbol{v}} = \frac{1}{K+1}\sum_{k=0}^{K}\boldsymbol{v}(t_k). \quad (5)$$

The more visible determinism is, the smaller E_{trans} will be. To reduce the stochastic errors associated with estimation, we seek the medians for 20 sets of 150 randomly chosen $\boldsymbol{x}(t_0)$ and then take the average over the 20 medians. Numerical work on random noises with various fractional power law indices indicated that for white (uncorrelated random) noise $E_{trans} \approx 1$ independently of D, while for colored noises E_{trans} monotonically decreases down to ~0.5 with D due to sustaining autocorrelation [4]. Figure 3 presents E_{trans} as a function of D. Determinism in the temperature sequence may be said to be more visible than expected for colored noises, which also suggests possible existence of chaos.

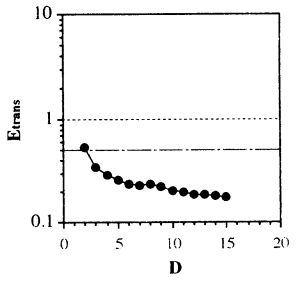


Figure 3. Wayland test on the the temperature sequence.

4 Time Series Prediction

Generalized radial basis function (RBF) networks [5] as predictors in the present work are of the form:

$$y(t + \tau\Delta t) = \sum_{h=1}^{M} w_h G[\boldsymbol{x}(t); \theta_h, \sigma_h] + \boldsymbol{a}\cdot\boldsymbol{x}(t) + b \quad (6)$$

$$G[\boldsymbol{x}(t);\theta_h,\sigma_h] = \exp(-\sigma_h \parallel \boldsymbol{x}(t) - \boldsymbol{\theta}_h \parallel^2) , \quad \frac{1}{1+\cosh(\sigma_h \parallel \boldsymbol{x}(t) - \boldsymbol{\theta}_h \parallel)}$$

where $\boldsymbol{a}$ and b are linear autoregressive (AR) parameters determined by least–mean–squared error fitting, w_h are weight parameters, σ_h and $\boldsymbol{\theta}_h$ are the widths and the centers of the basis functions, and M is the number of the basis functions. w_h, σ_h, and $\boldsymbol{\theta}_h$ are optimized so as to minimize the residuals of the linear AR predictive terms with stochastic gradient descent [6, 7]. The linear AR terms are supposed to reproduce autocorrelation of the time series, while the nonlinear terms are expected to capture nonlinear dynamics underlying the data. Although $(1+\cosh x)^{-1}$ belongs to the same class as Gaussians, it has a virtue in terms of hardware implementation that the exact shape of the input–output curve can be realized on a simple analog circuit system [7].

The networks have been trained for the library patterns generated from the first half of the time series under $M = 2, D = 3, \tau = 1, \Delta t = 10\ min$. To make predictions τ time steps into the future, we iterate the procedure that the output value of the predictor at a current time step is fed back to the input node at the subsequent time step. The predictive performances are summarized Table 1 where the predictive errors of a linear AR predictor and a trivial predictor $y(t+\tau\Delta t) = x(t)$ are shown for comparison.

Table 1. Predictive error as a function of the prediction time interval for the Gaussian RBF network, the $(1+\cosh x)^{-1}$ RBF network, the linear AR predictor, and a trivial predictor $y(t+\tau\Delta t) = x(t)$. Predictions are made for the second half of the data.

Predictive model			
Gaussian	$(1+\cosh x)^{-1}$	Linear AR	$x(t)$
$E(1)$ =0.134	0.133	0.133	0.166
$E(3)$ =0.324	0.325	0.329	0.422

Figure 4 shows $E(\tau)$ as a function of τ for the Gaussian RBF network, the $(1+\cosh x)^{-1}$ RBF network, and the linear AR predictor. For smaller τ there can be recognized little difference in the predictive performances between the linear predictor and the nonlinear predictors. In contrast, the predictive performance of the nonlinear predictors surpasses that of the linear predictor for larger τ. This may suggest that nonlinear determinism may lurk in the dynamics underlying the temperature sequence. Note that there is little difference in the predictive performance between the Gaussian and the $(1+\cosh x)^{-1}$ RBF networks as expected from the same class of the basis functions.

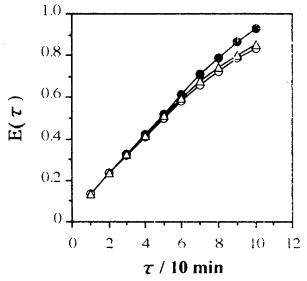


Figure 4. The predictive error as a function of the prediction time interval for the linear AR predictor (●), the Gaussian RBF network (○), and the $(1+\cosh x)^{-1}$ RBF network (△).

5 Summary

The present work shows that determinism can lurk in seemingly random sequences observed in actual systems and plants. Combinational use of distinct diagnostic algorithms for time series analysis is effective to avert misdiagnosis about dynamical properties of observational data. Adding a linear AR predictor to an RBF network seems to be a good strategy when handling complex data contaminated with observational noise.

Acknowledgment

We appreciate Ken Nakashima and Yasuharu Ikenaga for many stimulating discussions and technical support.

References

[1] T. Miyano, S. Kimoto, H. Shibuta, K. Nakashima, Y. Ikenaga, and K. Aihara, "Time series analysis and prediction on complex dynamical behavior observed in a blast furnace", *Physica* D, in print.

[2] G. Sugihara and R. M. May, "Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series", *Nature*, Vol.344, pp.734–741, 1990.

[3] R. Wayland, D. Bromley, D. Pickett, and A. Passamante, "Recognizing determinism in a time series", *Phys. Rev. Lett.*, Vol.70, pp.580–582, 1993.

[4] T. Miyano, "Time series analysis of complex dynamical behavior contaminated with observational noise", *Int. J. Bifurcation and Chaos*, Vol.6, pp.2031–2045, 1996.

[5] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures", *Neural Computation*, Vol.7, pp.219–269, 1995.

[6] T. Miyano and F. Girosi, "Forecasting global temperature variations by neural networks", *A. I. Memo*, No.1447, 1994 (Artificial Intelligence Laboratory, Massachusetts Institute of Technology).

[7] A. Nagami, H. Inada, and T. Miyano, "Generalized regularization networks with a particular class of bell-shaped basis function", *IEICE Trans. on Fundamentals*, in print.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Some examples on reconstructing nonlinear dynamics by the generalized exponential autoregressive model

Zhaoyun Shi, Yoshiyasu Tamura, Tohru Ozaki

The Institute of statistical Mathematics
4-6-7 Minami-Azabu, Minato-ku, Tokyo 106, Japan
zyshi@ism.ac.jp, tamura@ism.ac.jp, ozaki@ism.ac.jp

ABSTRACT: This study considers the problem of nonlinear time series analysis. We focus on the applications of the exponential autoregressive model. We introduce the original idea of the exponential autoregressive modeling at first. Then we expand the exponential autoregressive model to a generalized version. Finally, we give some examples on applications of the exponential autoregressive: feature extraction of machine tool chatter; reconstruction of quasi-periodic human pulse waves signal; reconstruction of spike and wave EEG time series.
Keywords: exponential autoregressive model, limit cycle, dynamics reconstruction

1. INTRODUCTION

Given a time series $\{x_t, t = 1,\ldots,N\}$, one of the important tasks of the traditional nonlinear time series modeling is to construct such a function, f: $R^p \rightarrow R^1$, with the form:

$$x_t = f(x_{t-1}, x_{t-2}, \ldots, x_{t-p}) + e_t \qquad (1)$$

where $f(\cdot)$ is an unknown nonlinear function and $\{e_t\}$ denotes noise usually regarded as a Gaussian white noise. In order to find such a unknown $f(\cdot)$ that maximizes the likelihood of the model, statisticians have provided several approaches, in which the exponential autoregressive (ExpAR) model[1] is famous one among the classic nonlinear time series models[2].

This study focuses on using the exponential autoregressive model to nonlinear time series analysis which usually means nonlinear time series prediction, characteristics analysis, feature extraction, and dynamics reconstruction. We investigate several possible applicability of both classic and generalized ExpAR model, and several application results are given to show the effectiveness of the model in extracting nonlinear dynamics.

2. EXPAR MODEL AND LIMIT CYCLE BEHAVIOR

One of the nonlinear phenomena studied in early time is limit cycle behavior which is beyond the capability of linear models. Generating limit cycle oscillation by differential equations was contributed by Duffing and Van der Pol at the early time of this century, while revealing limit cycle from time series by difference equation can find the contribution by Ozaki[1] who provided the ExpAR model as follows,

$$x_t = \sum_{i=1}^{p} (\varphi_i + \pi_i e^{-\gamma x_{t-1}^2}) x_{t-i} + e_t \qquad (2)$$

where γ is a scaling parameter. Note that the ExpAR model was provided originally by introducing the instantaneous roots in a linear framework to follow a nonlinear oscillation of limit cycle. The basic idea of the ExpAR model

is that: some roots of the model should be outside unit circle when the value of x_{t-1} near origin, then x_t starts to oscillate and diverges away from the origin; all roots should be inside the unit circle when the absolute value of x_{t-1} is sufficiently large, then x_t tends to be attracted towards the origin. Accordingly, Ozaki[1] had given the conditions of the existence of limit cycle behavior in the model: [a] All the roots of the characteristic equation:

$$\lambda^P - \varphi_1\lambda^{P-1} - \varphi_2\lambda^{P-2} - \cdots - \varphi_p = 0$$

lie inside the unit circle. Therefore x_t starts to damp out when $|x_{t-1}|$ becomes too large, while if the model satisfies the condition [b]: Some roots of the characteristic equation:

$$\lambda^P - (\varphi_1 + \pi_1)\lambda^{P-1} - \cdots - (\varphi_p + \pi_p) = 0$$

lie outside the unit circle, then x_t starts to oscillate and diverge for small $|x_{t-1}|$. The result of these two effects is expected to produce a similar sort of self-excited oscillation. Note that the above two conditions are necessary to produce a limit cycle but not sufficient. A sufficient condition for the existence of a limit cycle is [c]:

$$(1 - \sum_{i=1}^{p} \varphi_i) / \sum_{i=1}^{p} \pi_i > 1 \text{ or} < 0$$

Thus, by ExpAR modeling, we can achieve to know whether the underlying data is of limit cycle behavior or not just by checking the above three conditions. This will be helpful to design a monitoring system whenever the limit cycle be used as index.

3. GENERALIZED EXPAR MODEL

The classic ExpAR model has been accepted to be one of the useful nonlinear time series model[2]. However, we think it is not necessary to always use the classic ExpAR model to any complicated time series modeling, since the world we are faced with is so complex. We need much more general and powerful models to complex data analysis. When the classic ExpAR has been provided, someone also argue the assumption of coefficients depending only on the state x_{t-1} may limit the much wider applicability of the model since the coefficients may also depend on other states such as x_{t-j} in some cases. Here we consider generalization of the ExpAR model. Lets see the following general nonlinear dynamical model,

$$\frac{d^P x}{dt^P} + f_1(X)\frac{d^{P-1}x}{dt^{P-1}} + \ldots + f_{p-1}(X)\frac{dx}{dt} + f_p(X) = \varepsilon \tag{3}$$

without any loss of generality, we assume $X = (x, dx/dt, d^2x/dt^2, \ldots, d^{d-1}x/dt^{d-1})$ as the state-vector. When equation (3) is transformed into difference form, it becomes state-dependent AR model as the equation (4) with $X_{t-1} = (x_{t-1}, \Delta x_{t-1}, \Delta^2 x_{t-1}, \ldots, \Delta^{d-1} x_{t-1})$

$$x_t = \phi_0(X_{t-1}) + \sum_{i=1}^{p} \phi_i(X_{t-1}) x_{t-i} + e_t \tag{4}$$

State-dependent nonlinear time series model was provided by Priestley[2] as a kind of general model. However, problem remained in the model is how to specify the coefficients for an actual series. Here we consider the specification of the functional coefficients $\{\phi_i(X_{t-1}), i = 0,1,\ldots,p\}$ as a function learning problem, and therefore introduce Gaussian radial basis function (RBF) to specify the coefficients. Thus the coefficients of the model (4) can be written as the following form:

$$\phi_i(X_{t-1}) = c_{i,0} + \sum_{k=1}^{m} c_{i,k} \prod_{j=0}^{d-1} \exp\left(- \frac{(\Delta^j x_{t-1} - \Delta^j v_k)^2}{h_k} \right) \tag{5}$$

where $V_k = (v_k, \Delta v_k, \Delta^2 v_k, \ldots, \Delta^{d-1} v_k)$ denotes the centers corresponding to the original series and their difference sub-series. Actually this new model looks like a natural extension of the classic ExpAR model(2), we can prove that they share the same stable conditions

4. DYNAMICS EXTRACTION BY THE EXPAR MODEL

a) **Extract limit cycle behavior from data**

Can the ExpAR model be available for detecting limit cycle behavior from data? We use an example to answer the question. Figure 1 show two typical signal which were recorded at different stages of a cutting process. The signal "no chatter" denotes the data recorded at normal cutting state which is a normal vibration process; The signal "chatter" denotes the data recorded when chatter had been occurring in the cutting process. Since chatter is a self-excited oscillation of limit cycle type, the series "chatter" is expected to be of limit cycle behavior. Figure 2 are the unforced response of the ExpAR models estimated from the two series, which show the ExpAR models surely capture the true feature in this example.

b) **Quasi-periodic data reconstruction**

In nonlinear systems we can find quasi-periodicity very often. Some biomedical signal such as human pulse waves often act as quasi-periodicity. Someone also argue that it is difficult to distinguish the difference the quasi-periodicity and weak chaos, and quasi-periodicity is quite a common route to chaos. Therefore it is of importance in complex system analysis to develop any models available for reconstructing the dynamics of quasi-periodic time series. Here we try this problem by using the generalized exponential autoregressive model. The data used of typical quasi-periodicity is the recorded human pulse waves as shown in Figure 3. Figure 4 is the reconstructed series by the estimated GExpAR model from the original data, which show that the estimated model has really reconstructed the actual dynamics of the original data.

c) **Spike and wave data reconstruction**

Spike and wave activity can be regarded as a special case of the quasi-periodic time series, which is often seen in neuroscience field. It has been so far widely considered as a chaotic time series since the irregularities both in spike amplitudes and waves. Figure 5 shows an actual record of Epilepsy EEG of spike and wave activity. What we are interested is to investigate whether or not the ExpAR model can reconstruct this type of data. Figure 6 is a simulation of the estimated GexpAR model of the epilepsy EEG data, which shows the possibility of reconstructing the spike and wave complexity by the generalized ExpAR model.

5. CONCLUSION

This study introduced the exponential autoregressive models for nonlinear time series analysis. Some examples indicate that the ExpAR model is powerful for limit cycle extraction from original data. It is also shown that the generalized ExpAR model is available for dynamics reconstruction of quasi-periodic time series even when the series acts as the spike and wave. The future work should be concentrated in exploring much wider possible applications with improved GExpAR model estimation.

ACKNOWLEDGEMENTS

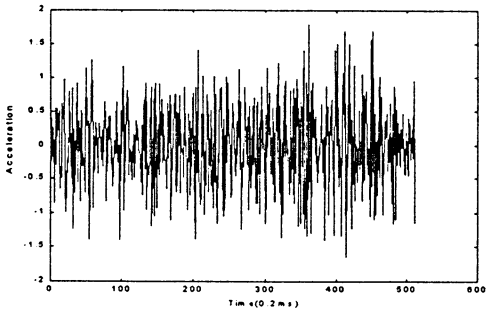
This research was supported by the Takahashi Science Foundation.

REFERENCES

[1] Ozaki, T (1985), Non-linear time series models and dynamical systems, *Time Series in the Time Domain, Handbook of Statistics*, **5**, Elsevier Science, North-Holland, pp. 25-83.

[2] Priestley, M B, (1988), *Non-linear and non-stationary time series analysis.* Academic Press, London.

(a) no chatter



(b) chatter

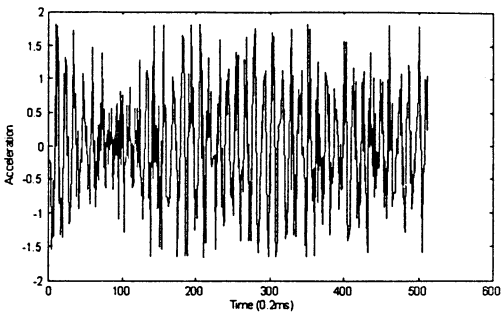
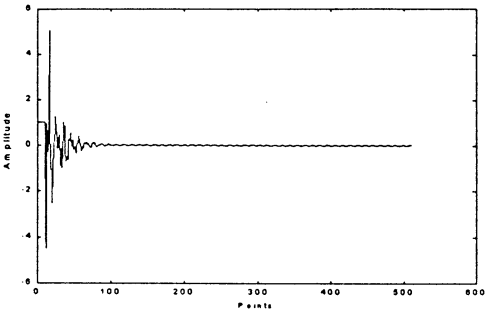


Figure 1 Records of cutting signal

(a) no chatter



(b) chatter

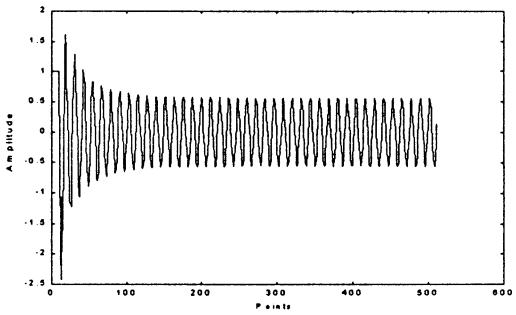


Figure 2 unforced response of the estimated ExpAR models

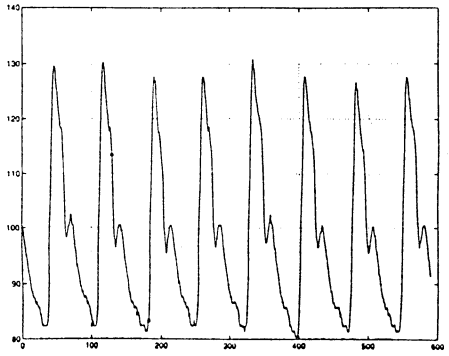


Figure 3 A record of human pulse waves

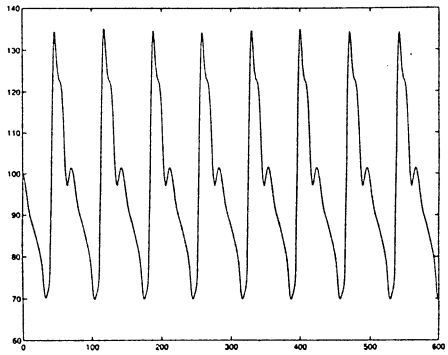


Figure 4 Unforced response of the GExpAR model estimated from the pulse waves series

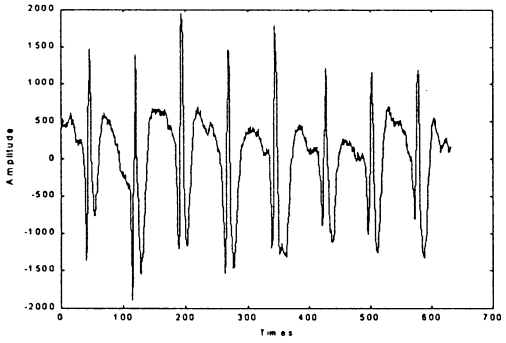


Figure 5 A record of epilepsy EEG data

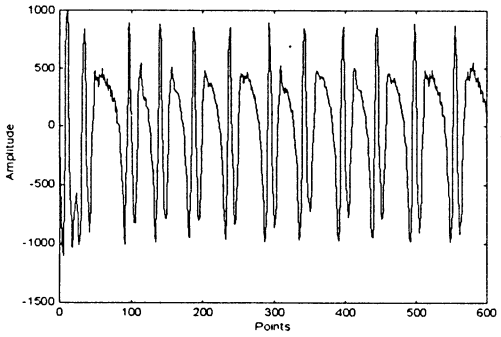


Figure 6 Unforced response of the GExpAR model estimated from the epilepsy EEG data

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Autonomous Distributed Control by Multi Agent Nets

Hirotomo Nakajima Toshiyuki Miyamoto Sadatoshi Kumagai

Department of Electrical Engineering
Osaka University
Suita 565-0871, JAPAN
e-mail : miyamoto{kumagai}@pwr.eng.osaka-u.ac.jp

Abstract

A multi agent net has been proposed as a formal description model for autonomous distributed systems(ADS's). In the model, a system is expressed by a set of agent nets, and each agent is an extended colored Petri net. This time we have developed the distributed runtime environment for the model. These extensions make it possible to simulate the model on distributed resources and acutually control the system with the model. This paper shows the software environment and explains how distributed simulations are executed.
Keywords: a multi agent net model, distributed runtime environment

1 Introduction

In recent years, autonomous distributed systems (ADS's) have been focused on a new system paradigm to cope with the complexity of large-scale computer embedded systems, such as manufacturing systems. An ADS is consisted of plural agents with an autonomous decision mechanism, and each agent acts on its own judgment in order to accomplish the purpose of the whole system. When we consider the control of a large-scale system, the ADS has the following advantages in comparison with traditional centralized control: robustness, flexibility and setup speed.

We have been developing a multi agent net model[3] as a formal description model for abstract and discrete event level of systems which consists of plural agents. A multi agent net consists of plural agent nets, and each agent net is an extended colored Petri net[1]. Behavior specifications and coordinating mechanisms of each agent are drawn on an agent net. In the multi agent net model, agent nets can communicate with each other, and thus an agent net can decide its behavior in cooperation with other agents.

We have developed the software environment of the model, a net editor and a simulator. However, they could only test behaviors of system described with the model, because they didn't support a distributed execution. Therefore we developed the distributed runtime environment of the model. This extension enables:

- distributing load of simulations among computers, and
- building distributed systems which actually control hardware.

In this paper we will focus mainly on the distributed simulation of the model. This paper is organized as follows; In Section 2, the multi agent net model are shown briefly. In Section 3, we show the distributed runtime environment, MAE. In Section 4, we explain the distributed simulation of the model. Section 5 concludes this paper.

2 MULTI AGENT NETS

In this section, we introduce a multi agent net model. In the multi agent net model, a system is actualized with a set of plural agent nets, and each agent net denotes one component of the system. Hereafter we will call a component an agent, and its net representation an agent net, whether or not it is autonomous. Actualizing a system with a set of agent nets makes it easy to change an assemblage of the system. In the multi agent net model, behavior of an agent is represented by a net structure of an agent net. An agent net is an extended colored Petri net(CPN). Figure 1 is an example of a multi agent net. The figure shows a simple communication protocol. That

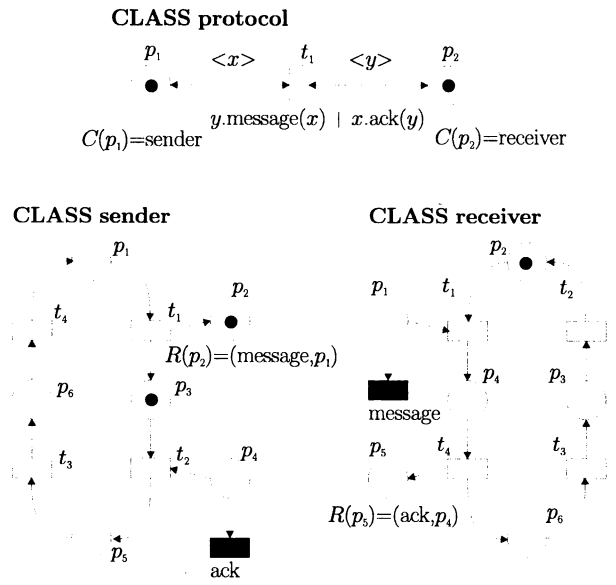


Figure 1: An example of agent nets.

is, a sender sends a message and a receiver replies. When there is one sender and one receiver, the system is represented by a multi agent net with three agent nets. The agent net with CLASS protocol on their left shoulder is the system, and manages the number of senders and receivers and their interactions. In the multi agent net, agent's interaction is effectuated by a communication among agent nets. That is called a rendezvous of agent nets. The agent net with CLASS sender and receiver describe the behavior of sender and receiver, respectively. Agent nets belonging to the same agent *class* have the same net structure, and an agent net is called an instance of the agent class. Each instance has its own net and state, and instances are differentiated by their own ID number. This will be explained further later in this paper.

One of differences between the multi agent net and ordinary CPN is that each agent class has its own color, and it can happen that an agent net takes the place of another agent net as a token. In Figure 1, the agent net **sender** has its color "sender", and the color specified to place p_1 in the agent net **protocol** is also "sender" from $C(p_1)$ = sender. That is, the token in the place p_1 in the net **protocol** is the net **sender**. In this case, the net **protocol** is called an upper-level net, and the **sender** is called a lower-level net. Please note that there is an agent net for each token that indicates an agent net, and the map need not be a bijection. Treating a net as a token enables us to change the number of agents easily. That is, by changing the number of tokens in the agent net **protocol**, we can change the number of senders and receivers arbitrarily.

Other difference between the multi agent net and CPN is that each agent net has two parts: an interface part and an implementation part, and other agent nets are allowed to access only the interface part. This access restriction technique is well know for encapsulation. In the interface part, methods provided by the agent net and their input places appear. Such a place is called an *input port* (of the method). In the implementation part, there are net structures without methods and arcs to methods. A method is fictitious and it is an action that is provided to other agent nets by the agent net. To execute the action, the agent must get tokens for input places of the method. In Figure 1, a method is represented by a black-filled box, and arcs from input places are drawn by dashed arrows. For example, the net **sender** provides a method ack, and the net **receiver** send an acknowledge message via this method. Please note that the implementation part is encapsulated, that is, designers of other agent nets can know only that the **sender** provides a method ack and that its input place is p_4. By using this encapsulation technique, one can lower the effects to other agent nets, which are caused by changing agent net structure. Please note that ordering relations with respect to method call should be provided in addition to methods and their input places, when an agent net has plural methods, because disregarding the method call order sometimes causes the deadlock of the system.

The multi agent net model supports reusing of design resources by using inheritances and aggregations of classes. The inheritance means that a new class inherits all the traits of old classes, and the aggregation means that a new class consists of plural agent classes. When class B inherits class A, class A is called a super-class of class B, and class B is a sub-class of class A. The inheritance of a class implies the inheritance of its color, namely we can say that "class B is class A". Therefore, when the classes have colors A and B respectively, a token can be in a place p_1 with $C(p_1) = A$, whether the token implies a net in class A or class B. The reverse, however, is not true. That is, when $C(p_2) = B$, a token that implies a net in class B can be in, but a token that implies a net in class A can not be in. The inheritance and the aggregation of a class are done by copying nets of the super-class. Here, although designers can change a part of the net even if the part is copied from the super-class, they can not remove methods and ordering relations with

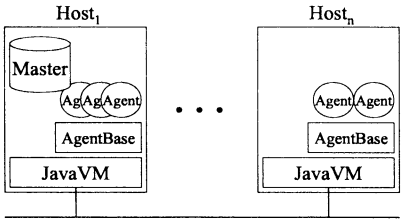


Figure 2: MAE runtime structure

respect to method calls.

3 MULTI AGENT NET LIBRARY

3.1 Multi Agents Environment

We have developed a Multi Agents Environment (MAE), see Figure 2, which is a distributed environment of handling general agents described in Java. MAE is consisted of three major components.

- **Agent**
  Agents of MAE can communicate with each other and migrate between hosts and have persistency. These Agents compose an autonomous distributed control system.

- **AgentBase**
  We assume that there is a unique AgentBase on each host. AgentBases are bound by BaseMaster at their creation. AgentBases provide the following services for agents: communication, migration and preservation. We can easily find which Agents are on the host.

- **BaseMaster**
  Currently, we assume that there is only one BaseMaster on MAE. It provides agent-id-services: e.g., creating agent id, searching where the agent referred by an agent id is, and so on. It also manages AgentBases.

3.2 An Implementation of Multi Agent Nets

We have produced a library for the multi agent net on MAE, so that we can construct systems more flexibly. For example, we can simulate a system which is consisted of agents written in the multi agent net model and written in Java. See Section 4.

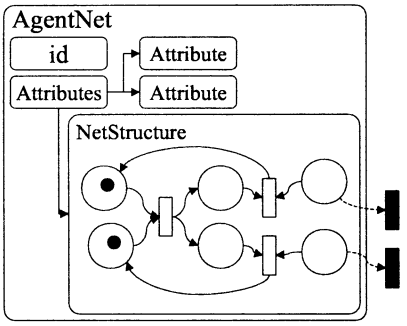


Figure 3: A sample AgentNet structure

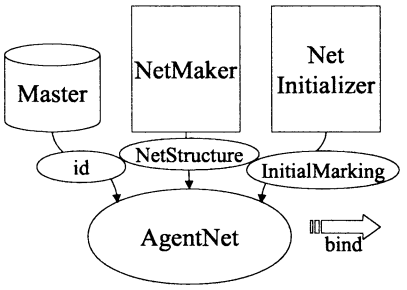


Figure 4: An AgentNet creation

Figure 3 shows a general agent net structure. An AgentNet class is derived from MobileAgent class. An instance of AgentNet has some attributes, for example, a list of collaborating fellows, and tasks to execute. A NetStructure is one of the attributes which are characteristic of AgentNet and storage NetComponents, Transition, Place, Method and so on. AgentNets are capsulated completely, except for Method, which simplifies the interaction between AgentNets.

In the following, we explain how an AgentNet is created, see Figure 4. To create an AgentNet, we have to execute the following steps:

- **An agent id created by BaseMaster**
  Each Agent has a unique id given at its birth and BaseMaster manages Agents with the id.

- **NetMaker class**
  An AgentNet has a NetStructure as one of its attributes. NetMaker class constructs a NetStructure.

- **NetInitializer class**
  NetInitializer class initializes the state of an AgentNet as initial markings.

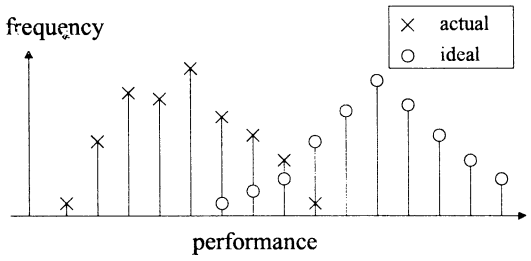


Figure 5: Ideal performances as measure

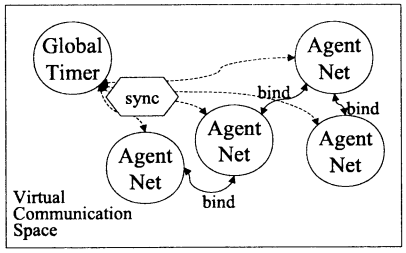


Figure 6: Simulation for ideal performance

4 A DISTRIBUTED SIMULATION WITH THE MULTI AGENT NET MODEL

We prepared two types of simulations for our purposes. We prepared the simulation for evaluation of ideal performances of systems described with the multi agent net model. Also, we prepared the simulation for evaluation of actual performances of systems. Note that both of these studies are experimental and not analytical.

4.1 An Evaluation of Ideal Performance

Here, an 'ideal' performance means the performance when we assume that all processes are always executed in a pre-defined span, regardless of facilities of hosts composing the simulator. We can use an ideal performance as a measure in tuning the system, Figure 5 is an image of utilization of ideal performances.

To do this, we designed '*GlobalTimer*' agent, which makes all AgentNets synchronized, so that they are executed in the virtual-time. Here, the span is defined as the preparing time for firing and binding the transitions and tokens. Figure 6 shows the runtime structure of this simulation.

4.2 An Evaluation of Actual Performance

Otherwise, 'actual' means that the simulation is executed in real-time, not virtual-time. This simulation can test if the model can actually cooperate with physical devices. This is very important because it is generally so difficult for an asynchronous distributed system to cooperate with physical devices because the system cannot guarantee to perform in real-time.

In order to check its synchronization, we have enhanced logging modules of each NetComponent, Transition, Place and Token. Thus we can do this by tracing those histories.

5 CONCLUSION

This paper introduced the distributed runtime environment and two formed distributed simulations with the multi agent net model. This enhancement gives the distributed systems design with the model a lot of utility value. We are going to incorporate these tools into the Integrated Developing Environment for the system.

Acknowledgements

Part of this research is supported by Intelligent Manufacturing System(IMS) project.

References

[1] K. Jensen, "Coloured Petri Nets," *Spring-Verlag*, 1992.

[2] T. Murata, "Petri Nets : Properties, analysis and applications," *Proceedings IEEE,* Vol.77, No.4, pp.541-580, 1989.

[3] T. Miyamoto and S. Kumagai, "A Multi Agent Net Model of Autonomous Distributed Systems," *Proceedings of CESA'96, Symposium on Discrete Events and Manufacturing Systems*, pp.619-623, 1996.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Estimate Based Limited Lookahead Control of Discrete Event Systems with Model Uncertainty

Shigemasa Takai
Department of Opto-Mechatronics
Wakayama University
930 Sakaedani, Wakayama 640-8510, Japan

Abstract

This paper assumes that it is only known that the exact model of the plant belongs to a set of possible models. The control objective is to synthesize a nonblocking supervisor under such model uncertainty. The estimate based limited lookahead supervisor proposed by Kumar *et al.* is used to solve the problem. It is shown that if there is no starting error in the closed-loop system, then the estimate based limited lookahead supervisor is a nonblocking supervisor for the plant with model uncertainty considered in this paper.

Key words. discrete event system, supervisory control, model uncertainty, limited lookahead policy, estimate

1 Introduction

Ramadge and Wonham [4] have proposed the supervisory control theory as a theoretical framework for controlling discrete event systems (DESs). Most research on supervisory control assumes that the exact model of a DES to be controlled, called a plant, is available. The conventional supervisory control problem is to synthesize a controller, called a supervisor, such that it achieves legal behavior for the plant model. Ramadge and Wonham [4] have presented necessary and sufficient conditions for the existence of a nonblocking supervisor for the marked language specification.

In this paper, we assume that we only know that the exact model G of the plant belongs to a set of possible models $\{G_i | \ i \in I\}$ where I is an index set. Lin [3] has proposed an approach, called the adaptive supervision, to control a plant with model uncertainty under the assumption that the legal marked behavior K of G is a sublanguage of all possible plant languages. In this approach, the set of possible models is updated based on the previously executed strings in the plant in order to reduce model uncertainty. More precisely, after a string s has occurred in the plant, models which cannot execute the string s are eliminated from the set of possible models. When the set of possible models is updated, the supervisor is reconfigured based on the updated set.

Let K_{legal} be the specification language which consists of all legal marked strings. This paper assumes that the legal marked behavior of the plant G is given as $K = K_{legal} \cap L_m(G)$ [2], where $L_m(G)$ is the marked language of G. The control objective is to synthesize a nonblocking supervisor for the legal marked behavior K. Note that K is not necessarily a sublanguage of all possible plant languages. Therefore, the adaptive supervision proposed by Lin [3] cannot be applied to our problem in general. This paper uses the estimate based limited lookahead supervisors [2] under the assumption that an N-step projection of the future behavior of G is available at each step. The estimate based limited lookahead supervisor determines the next control action based on an N-step projection of the plant behavior and an estimate of the plant behavior beyond the projection. Note that the estimate based limited lookahead supervisor is, in general, more permissive than the conservative limited lookahead supervisor proposed by Chung *et al.* [1]. It is shown that if there is no starting error [1], [2] in the closed-loop behavior, then the estimate based limited lookahead supervisor is a nonblocking supervisor for the plant G with model uncertainty considered in this paper. This result complements the work of [2] on the estimate based limited lookahead supervision.

The robust supervision [3] can be also applied to our problem. The robust supervisor is synthesized *off-line* by using the entire models of the possible plant behaviors and the specification, while the estimate based limited lookahead supervisor is synthesized *on-line* by using an $N + 1$-step projection of the estimate of the plant behavior and an N-step projection of the specification. When the behaviors of possible models are

complex, the estimate based limited lookahead supervisor is synthesized more efficiently than the robust supervisor.

2 Preliminaries

Let Σ be the set of events which occur in a DES G to be controlled, called a plant. We use Σ^* to denote the set of all finite strings of elements in Σ, including the empty string ε. Let $L \subseteq \Sigma^*$ be a language. We denote the set of all prefixes of strings in L by $pr(L)$. Given a string $s \in \Sigma^*$, the language L *after* s, denoted by $L\backslash s$, is defined as $L\backslash s = \{t \in \Sigma^* | \ st \in L\}$. Let $\mathcal{N}$ be the set of all nonnegative integers. For any string s, $|s| \in \mathcal{N}$ denotes the length of s. The *truncation* of L to $N \in \mathcal{N}$, denoted by $L|_N$, is defined as $L|_N = \{t \in L| \ |t| \leq N\}$.

Let $G := (L(G), L_m(G))$ be the language model of the plant G, where $L_m(G) \subseteq L(G) = pr(L(G)) \subseteq \Sigma^*$. $L(G)$ and $L_m(G)$ are the languages generated and marked by G, respectively. Let $L_{complete} \subseteq \Sigma^*$ be the set of all strings which correspond to completion of a certain task. We assume that $L_m(G)$ is obtained as $L_m(G) = L(G) \cap L_{complete}$ [2]. Let $G\backslash s := (L(G)\backslash s, L_m(G)\backslash s)$ for each $s \in L(G)$, and $G|_N := (L(G)|_N, L_m(G)|_N)$ for each $N \in \mathcal{N}$. $G\backslash s$ represents the behavior of the plant G after s has occurred in G. $G|_N$ represents the N-step projection of the behavior of G. For any two language models $G_1 := (L(G_1), L_m(G_1))$ and $G_2 := (L(G_2), L_m(G_2))$,

$$G_1 \leq G_2 \Leftrightarrow [L(G_1) \subseteq L(G_2)] \wedge [L_m(G_1) \subseteq L_m(G_2)].$$

$G_1 \leq G_2$ implies that the behavior of G_1 is included in the behavior of G_2.

As in the usual supervisory control framework [4], the event set Σ is decomposed into two subsets Σ_c and Σ_u of controllable and uncontrollable events, respectively, where $\Sigma_c \cap \Sigma_u = \emptyset$. Let $L \subseteq L_m(G)$ be a language. L is said to be

- controllable with respect to G [4] if

$$pr(L)\Sigma_u \cap L(G) \subseteq pr(L).$$

- relative-closed with respect to G [4] if

$$pr(L) \cap L_m(G) = L.$$

We use $\mathcal{RC}(L, G)$ to denote the set of relative-closed and controllable sublanguages of $L \subseteq L_m(G)$ with respect to G. There always exists its supremal element $\sup \mathcal{RC}(L, G)$, called the supremal relative-closed and controllable sublanguage of L with respect to G [6].

Let $K_{legal} \subseteq \Sigma^*$ be the set of all legal strings. We assume that the legal marked behavior $K \subseteq L_m(G)$ for G is obtained as $K = K_{legal} \cap L_m(G)$ [2]. This paper considers estimate based limited lookahead supervisors [2], [5] which determine the next control action based on not only an N-step ($N \in \mathcal{N}$) projection of the plant behavior but also an estimate of the plant behavior beyond the projection. Note that original limited lookahead supervisors proposed by Chung *et al.* [1] use only an N-step projection of the plant behavior to determine the next control action. We define an estimate $E^N(L(G)\backslash s)$ of the generated language $L(G)\backslash s$ after $s \in L(G)$ as a language satisfying the following conditions [5].

- $E^N(L(G)\backslash s)$ is closed.
- $E^N(L(G)\backslash s)|_N = L(G)\backslash s|_N$.

The estimate of the marked language $L_m(G)\backslash s$ after s, denoted by $E^N(L_m(G)\backslash s)$, is defined as follows [2]:

$$E^N(L_m(G)\backslash s) = E^N(L(G)\backslash s) \cap \{L_{complete}\backslash s\}.$$

Let $E^N(G\backslash s) := (E^N(L(G)\backslash s), E^N(L_m(G)\backslash s))$. The estimate based limited lookahead supervisor $\gamma^N : L(G) \to 2^\Sigma$ proposed in [2] is defined as

$$\gamma^N(s) = \{pr(\sup \mathcal{RC}(K\backslash s|_N, E^N(G\backslash s)|_{N+1})) \cap \Sigma\} \cup \Sigma_u.$$

γ^N is computed by using the N-step projection $K\backslash s|_N$ of the legal behavior and the $N+1$-step projection $E^N(G\backslash s)|_{N+1}$ of the estimate of the plant behavior after each $s \in L(G)$ has occurred. The language generated in the closed-loop system with γ^N, denoted by $L(G, \gamma^N)$, is defined as follows:

- $\varepsilon \in L(G, \gamma^N)$,
- $(\forall s \in L(G, \gamma^N)$ and $\forall \sigma \in \Sigma)$

$$s\sigma \in L(G, \gamma^N) \Leftrightarrow [s\sigma \in L(G)] \wedge [\sigma \in \gamma^N(s)].$$

The marked language in the closed-loop system, denoted by $L_m(G, \gamma^N)$, is defined as $L_m(G, \gamma^N) = L(G, \gamma^N) \cap L_m(G)$. The supervisor γ^N is said to be nonblocking if $pr(L_m(G, \gamma^N)) = L(G, \gamma^N)$.

The following two properties on the estimate $E^N(L(G)\backslash s)$ $(\forall s \in L(G))$ have been defined in [2].

P1: $E^{N+1}(L(G)\backslash s) \subseteq E^N(L(G)\backslash s)$.

P2: $E^{N+1}(L(G)\backslash s)\backslash \sigma = E^N(L(G)\backslash s\sigma)$ provided that $s\sigma \in L(G)$ $(\forall \sigma \in \Sigma)$.

Kumar *et al.* [2] have shown that if $E^N(L(G)\backslash s)$ satisfies both P1 and P2 for all $s \in L(G)$, then γ^N is a nonblocking supervisor under some assumption.

Proposition 1 [2] *Assume that the estimate $E^N(L(G)\backslash s)$ satisfies P1 and P2 for all $s \in L(G)$. If $\sup \mathcal{RC}(K\backslash\varepsilon|_N, E^N(G\backslash\varepsilon)) \neq \emptyset$, then γ^N is a nonblocking supervisor such that $L_m(G,\gamma^N) \subseteq K$.*

If $\sup \mathcal{RC}(K\backslash\varepsilon|_N, E^N(G\backslash\varepsilon)) \neq \emptyset$, then we say that there is no starting error in $L(G,\gamma^N)$ [1], [2].

3 Control of DESs with Model Uncertainty

This section considers a situation where we only know that the exact model G of the plant belongs to a set of possible models

$$\mathcal{G} = \{G_i |\ i \in I\},$$

where I is an index set [3]. The control objective is to synthesize a nonblocking supervisor for G under such model uncertainty. We use the estimate based limited lookahead supervisor to solve the problem under the assumption that we exactly know the N-step projection $L(G)\backslash s|_N$ of the plant behavior after each string $s \in L(G)$ has occurred in the plant.

Lin [3] has proposed an procedure, called an identification procedure, to reduce uncertainty by eliminating some models from $\mathcal{G}$. We modify the Lin's procedure by using the N-step projection of the future behavior. For each $s \in L(G)$, we use $\mathcal{G}(s)$ to denote the set of possible models after s has occurred. The set $\mathcal{G}(\cdot)$ is defined inductively as follows:

- $\mathcal{G}(\varepsilon) = \{G_i \in \mathcal{G} |\ L(G_i)\backslash\varepsilon|_N = L(G)\backslash\varepsilon|_N\}$.
- $(\forall s\sigma \in L(G)\ (\sigma \in \Sigma))$

  $\mathcal{G}(s\sigma) = \{G_i \in \mathcal{G}(s) |\ L(G_i)\backslash s\sigma|_N = L(G)\backslash s\sigma|_N\}$.

Each element $G_i \in \mathcal{G}(s)$ has the same N-step projection of the future behavior as the exact model G has. Note that if $N = 0$, then our procedure is the same as the Lin's procedure.

After a string $s \in L(G)$ has occurred, the information that we have is that the exact model G belongs to $\mathcal{G}(s)$. In other words, we only know that a string in $\bigcup_{G_i \in \mathcal{G}(s)} L(G_i)\backslash s$ will occur after s. Using this information, an estimate $E^N(L(G)\backslash s)$ $(\forall s \in L(G))$ can be obtained as

$$E^N(L(G)\backslash s) = \bigcup_{G_i \in \mathcal{G}(s)} L(G_i)\backslash s. \qquad (1)$$

We use the estimate $E^N(L(G)\backslash s)$ defined by (1) to synthesize the estimate based limited lookahead supervisor γ^N. Note that $E^N(L(G)\backslash s)$ defined by (1) does not necessarily satisfy P2 as shown in the following example. So the results of [2] cannot be applied directly to our problem.

Example 1 Let $\Sigma = \{a, b, c\}$ and $L_{complete} = \{a, aa, aaa\}$. We consider two possible models $G_1 = (L(G_1), L_m(G_1))$ and $G_2 = (L(G_2), L_m(G_2))$ where

$$L(G_1) = pr(\{aaa, bc\}),\ L_m(G_1) = \{a, aa, aaa\}$$

and

$$L(G_2) = pr(\{aac, bb\}),\ L_m(G_2) = \{a, aa\}.$$

The automata representations of G_1 and G_2 are shown in Figs. 1 and 2, respectively. Assume that $G := G_1$ is the exact model.

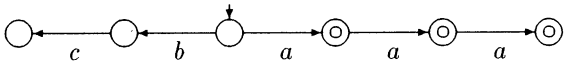


Fig. 1. A possible model G_1.

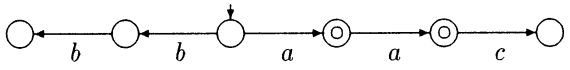


Fig. 2. A possible model G_2.

First, we consider the case that $N = 1$. Since $L(G_1)\backslash\varepsilon|_1 = L(G_2)\backslash\varepsilon|_1 = pr(\{a, b\})$ and $L(G_1)\backslash a|_1 = L(G_2)\backslash a|_1 = pr(\{a\})$, we have

$$\mathcal{G}(\varepsilon) = \mathcal{G}(a) = \{G_1, G_2\}.$$

It follows that

$$\begin{aligned} E^1(L(G)\backslash a) &= (L(G_1)\backslash a) \cup (L(G_2)\backslash a) \\ &= pr(\{aa, ac\}). \end{aligned}$$

We next consider the case that $N = 2$. Since $L(G_1)\backslash\varepsilon|_2 = pr(\{aa, bc\})$ and $L(G_2)\backslash\varepsilon|_2 = pr(\{aa, bb\})$, we have $\mathcal{G}(\varepsilon) = \{G_1\}$. It follows that

$$\begin{aligned} E^2(L(G)\backslash\varepsilon)\backslash a &= (L(G_1)\backslash\varepsilon)\backslash a \\ &= pr(\{aa\}). \end{aligned}$$

Thus, $E^2(L(G)\backslash\varepsilon)\backslash a \subset E^1(L(G)\backslash a)$, which implies that P2 does not hold for $N = 1$.

We define another property on the estimate $E^N(L(G)\backslash s)$ $(\forall s \in L(G))$ as follows:

P3: $L(G)\backslash s \subseteq E^N(L(G)\backslash s)$.

P3 implies that the estimate $E^N(L(G)\backslash s)$ includes the actual behavior of the plant after $s \in L(G)$. Obviously, $E^N(L(G)\backslash s)$ defined by (1) satisfies P3 for all $s \in L(G)$.

The following theorem shows that if $E^N(L(G)\backslash s)$ satisfies P3 (instead of P1 and P2) for all $s \in L(G)$, then γ^N is a nonblocking supervisor under the same assumption as Proposition 1.

Theorem 1 *Assume that the estimate $E^N(L(G)\backslash s)$ satisfies P3 for all $s \in L(G)$. γ^N is a nonblocking supervisor such that $L_m(G,\gamma^N) \subseteq K$ if there is no starting error in $L(G,\gamma^N)$.*

Kumar *et al.* have proved that if $E^N(L(G)\backslash s)$ satisfies P1 and P2 for all $s \in L(G)$, then γ^N is a nonblocking supervisor under the assumption of the absence of the starting error in $L(G,\gamma^N)$ [2]. Theorem 1 shows that this result is also true when we assume P3 instead of P1 and P2. So Theorem 1 complements the work of [2].

Example 2 Again, we consider the two models G_1 and G_2 in Example 1. Let $\Sigma_c = \{a,b\}$, $\Sigma_u = \{c\}$ and $K_{legal} = L_{complete} = \{a, aa, aaa\}$. As in Example 1, we assume that $G = G_1$ is the exact model. So $K = \{a, aa, aaa\}$.

Consider the case that $N = 1$. Since $K\backslash\varepsilon|_1 = \{a\}$ and $E^1(G\backslash\varepsilon) = (L(G_1) \cup L(G_2), L_m(G_1) \cup L_m(G_2))$, we have

$$\sup \mathcal{RC}(K\backslash\varepsilon|_1, E^1(G\backslash\varepsilon)) = \{a\} \neq \emptyset, \qquad (2)$$

which implies that $\gamma^1(\varepsilon) = \{a,c\}$. So only a can occur at the initial state in the closed-loop system. By Theorem 1 and (2), γ^1 is a nonblocking supervisor for G. Also, since $K\backslash a|_1 = pr(\{a\})$ and $E^1(G\backslash a) = ((L(G_1) \cup L(G_2))\backslash a, (L_m(G_1) \cup L_m(G_2))\backslash a)$, we have

$$\sup \mathcal{RC}(K\backslash a|_1, E^1(G\backslash a)) = \{\varepsilon\},$$

which implies that $\gamma^1(a) = \{c\}$. So no event can occur in the closed-loop system after a has occurred. Therefore, the marked behavior achieved by γ^1 is $L_m(G,\gamma^1) = \{a\} \subseteq K$.

4 Conclusions

This paper has considered the situation where we only know that the exact model of the plant belongs to a set of possible models. Under the situation, the control problem is synthesizing a supervisor to control the plant with such model uncertainty. We have applied the estimate based limited lookahead supervision proposed by Kumar *et al.* [2] to the problem.

We have modified the Lin's identification procedure [3] to obtain the estimates of the future behavior. Then we have proved that if there is no starting error in the closed-loop behavior, then the estimate based limited lookahead supervisor based on the obtained estimates is a nonblocking supervisor for the plant with model uncertainty considered in this paper. This means that the estimate based limited lookahead supervision is a useful technique for controlling a plant with model uncertainty.

Acknowledgement

This work was supported by Grant-in-Aid for Encouragement of Young Scientists (No. 09750502) from The Ministry of Education, Science, Sports and Culture of Japan.

References

[1] S. L. Chung, S. Lafortune and F. Lin, "Limited lookahead policies in supervisory control of discrete event systems," *IEEE Trans. Automat. Contr.*, Vol. 37, No. 12, pp. 1921–1935, 1992.

[2] R. Kumar, H. M. Cheung and S. I. Marcus, "Extension based limited lookahead supervision of discrete event systems," *Automatica*, to appear.

[3] F. Lin, "Robust and adaptive supervisory control of discrete event systems," *IEEE Trans. Automat. Contr.*, Vol. 38, No. 12, pp. 1848–1852, 1993.

[4] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete-event processes," *SIAM J. Contr. Optim.*, Vol. 25, No. 1, pp. 206–230, 1987.

[5] S. Takai, "Estimate based limited lookahead supervisory control for closed language specifications," *Automatica*, Vol. 33, No. 9, pp. 1739–1743, 1997.

[6] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language," *SIAM J. Contr. Optim.*, Vol. 25, No. 3, pp. 637–659, 1987.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Conflict Resolution in Continuous Petri Nets Using Linear Programming

A. Tanaka T. Ushio
Dept. of Systems and Human Science
Graduate school of Engineering Science
Osaka University
1-3 Machikaneyama, Toyonaka
Osaka 560-8531, Japan

S. Kodama
Dept. of Electronics
Graduate school of Engineering
Kinki University
3-4-1 Kowakae, Higashi-Osaka
Osaka 577-0818, Japan

Abstract

Timed continuous Petri nets (TCPNs) have been proposed as an approximation of behavior of timed Petri nets with a large number of reachable markings. In TCPNs, however, behavior due to a firing rule called weak enabling cannot be well-defined. So we propose a new firing rule introducing the concept of delay time of firing. Next, we consider a conflict resolution problem and introduce a resolution policy that we use as many tokens as possible for firing subject to preempted enabling transitions, formulated by a linear programming problem. Thus the conflict resolution can be achieved systematically.

Keywords: continuous Petri nets, modeling, firing rules, conflict, linear programming

1 Introduction

Timed continuous Petri nets (TCPNs) [1] have been proposed as an approximation of behavior of timed Petri nets (TPNs) [2] with a large number of reachable markings by allowing that tokens can take real numbers. Firing of transitions is continuous in TCPNs whereas it is instantaneous in "conventional" TPNs or timed discrete PNs. Firing speeds of enabling transitions in TCPNs are obtained by an approximation method of changes of markings. Continuous firing of a transition is possible if all input places to the transition which have no tokens have a firing input transition, which is a very unique feature in TCPNs. It is called that such a transition is weakly enabled.

However, an introduction of weakly enabled transitions implies that relationship between time elapse mechanism and change of markings becomes ambiguous. So, we show that, because of the ambiguity, a circle of weakly enabled transitions causes indeterminacy of their firing speeds. Thus, we propose a modification of firing rules in order to remove such indeterminacy. By the modified rules, we represent delay time for move of tokens explicitly.

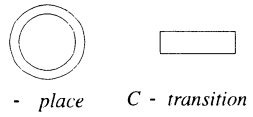


Figure 1: Nodes of CPNs

Next, we consider a conflict resolution problem in TCPNs. A simple conflict consisting of one place and two transitions can be resolved by introducing a straightforward extension of well-known conflict resolution rules in TPNs. However, for more complex conflict structures, we need new conflict resolution rules since weak enabling should be taken into account. We introduce a resolution policy that we use as many tokens as possible for firing subject to preempted enabling transitions, and show that this policy is formulated by a LP problem. Thus the conflict resolution can be achieved systematically.

2 Continuous Petri nets

Structures of a PN [2] is a directed, weighted, bipartite graph consisting of two kinds of nodes, called places and transitions. In this paper, a place and a transition are represented graphically as in Fig. 1.

Depending on firing rules, many variations of TCPNs have been proposed. In this paper, we use a constant speed continuous Petri net (CCPN), and it is easy to extend the following discussion to other TCPNs [1].

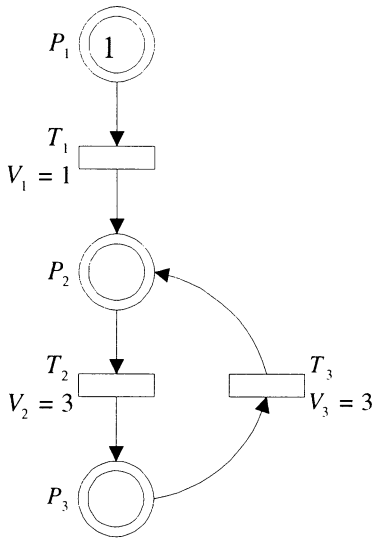


Figure 2: A TCPN with the circle of weakly enabled transitions

Definition 1 *A Constant speed continuous Petri net is represented by*

$$CCPN = \langle P, T, I, O, Tempo, M_0 \rangle \qquad (1)$$

where P is a finite set of continuous (C) places, T is a finite set of continuous (C) transitions, $P \cap T = \emptyset$, and $P \cup T \neq \emptyset$. $I : P \times T \to \Re^+ \cup \{0\}$ represents input arcs with the weight of nonnegative real number. $O : P \times T \to \Re^+ \cup \{0\}$ represents output arcs with the weight of nonnegative real number. $Tempo : T \to \Re^+$ is a function associating a positive real number to each transition, and $Tempo(T_j) = V_j$ represents a maximum firing speed of T_j. M_0 is an initial marking containing nonnegative real numbers. For each $d \in P \cup T$, ${}^\bullet d$ and $d^\bullet$ indicate a set of input nodes of d and a set of output nodes of d, respectively. A C transition can be either strongly enabled or weakly enabled, depending on markings of its input places.

Definition 2 (The firing rules of a C transition)

1. *If either a C transition T_j is a source transition or $M(P_i) > 0$ for each input C place $P_i \in {}^\bullet T_j$, T_j is said to be "strongly enabled" at time t and then fire with the maximum firing speed $v_j(t) = V_j$.*

2. *If, for each input C place $P_i \in {}^\bullet T_j$ with $M(P_i) = 0$, there exists its input C transition $T_j \in {}^\bullet P_i$ such that T_j fires, T_j is said to be "weakly enabled" at time t and then fire weakly with the firing speed in accordance with the firing speeds of the upstream C transitions.*

3. *If T_j is enabled at time t, T_j can fire continuously, and for sufficiently small $\delta > 0$, $I(P_i, T_j) \times \int_t^{t+\delta} v_j(t)dt$ tokens are removed from each input C place $P_i \in {}^\bullet T_j$ and $O(P_i, T_j) \times \int_t^{t+\delta} v_j(t)dt$ tokens are added to each output C place $P_i \in T_j^\bullet$.*

3 A modified timed continuous Petri net

We consider the behavior of a CCPN shown in Fig. 2. Transition T_1 is strongly enabled at an initial state ($t = 0$), while transitions T_2 and T_3 are weakly enabled at the same time due to the input places of these transitions without tokens. The instantaneous firing speed of T_2 depends on that of T_3, and vice versa, since the firing speeds of weakly enabled transitions depend on the firing speed of upstream enabled transitions. Thus the firing speeds of T_2 and T_3 are not determined. Hence the behavior of Fig. 2 is not well-defined. In other words, delay time is not associated to C transitions which fire continuously and the instantaneous firing speeds of the weakly enabled C transitions at time t is determined by the instantaneous firing speeds, at the same time t, of the upstream C transitions of the weakly enabled C transtion. So, in the directed circuit consisting of C transitions and C places without tokens, the instantaneous firing speeds are not determined.

In order to avoid such a situation, we introduce delay time to the firing of C transitions. Thus markings are represented by $M = M^r + M^n$, where M^r and M^n are a reserved marking and a nonreserved marking, respectively.

Definition 3 *A modified constant speed continuous Petri net is represented by*

$$\langle CCPN, Delay \rangle \qquad (2)$$

where $Delay : T \to \Re^+$ is a function associating a nonnegative real number to each C transition, and $Delay(T_j) = d_j$ represents delay time of T_j.

Definition 4 *(The firing rules of a timed C transition)*

1. *If either a C transition T_j is a source transition or $M^n(P_i) > 0$ for each input C place $P_i \in {}^\bullet T_j$, T_j*

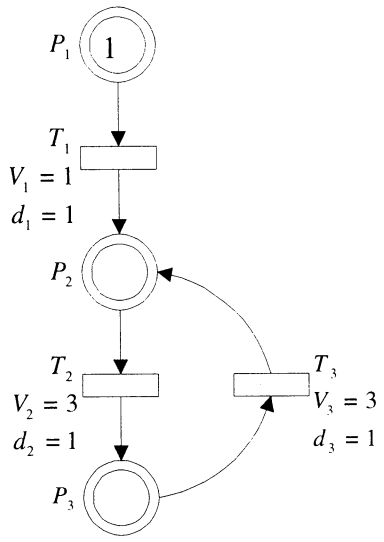


Figure 3: The modified TCPN with the circle of weakly enabled transitions

is said to be "strongly enabled" at time t and then fire with the maximum firing speed $v_j(t) = V_j$.

2. *If, for each input C place* $P_i \in {}^\bullet T_j$ *with* $M(P_i) = 0$, *there exists its input C transition* $T_j \in {}^\bullet P_i$ *such that* T_j *fires,* T_j *is said to be "weakly enabled" at time t and then fire weakly with the firing speed in accordance with the firing speeds of the upstream C transitions.*

3. *Enabled* T_j *reserves nonreserved tokens in each input C place. If* T_j *keeps on reserving nonreserved tokens in each input C place during time interval* $[t, t+\delta]$, $I(P_i, T_j) \times \int_t^{t+\delta} v_j(t)dt$ *nonreserved tokens are reserved in each input C place* $P_i \in {}^\bullet T_j$.

4. *Tokens reserved at time t are removed from each input C place of* T_j *at time* $t + d_{T_j}$, *and nonreserved tokens are added to each output C place of* T_j *at time* $t + d_{T_j}$.

Associated the delay time $d_j = 1, j = 1, \ldots, 3$ to the C transitions in Fig. 2, we obtain the modified Petri net as in Fig. 3, and the behavior of Fig. 3 is shown in Fig. 4. Hence clearly, associated delay to firing of each C transition, behavior in a directed circuit consisting of C transitions and C places without tokens is well-defined at each time.

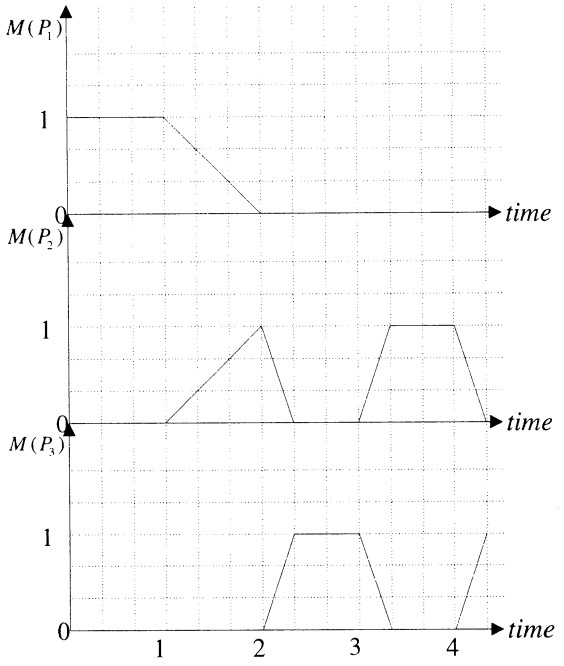


Figure 4: The behavior of the modified TCPN

4 Firing speeds and conflicts of weakly enabled transitions

Regardless of existence of structural conflicts, effective conflicts can't occur for strongly enabled transitions firing with the maximum speed, but occur for weakly enabled transitions due to the instantaneous firing speeds accordance with the rate of tokens provided in the input C places without nonreserved tokens. There are two kinds of calculation of instantaneous firing speeds of weakly enabled transitions, for a single input C place(Fig. 5) and for more than one input C places(Fig. 6), without nonreserved tokens.

We consider cases that transitions become weakly enabled at time t. Let a set of weakly enabled output transitions of C place P_i such that $M^n(P_i) = 0$ be $T_{P_i}^{weak}$, the maximum firing speed of the each weakly enabled transition $T_j^{weak} \in T^{weak}(P_i)$ be V_j^{weak}, the instantaneous firing speed at time t be $v_j^{weak}(t)$, a set of enabled input C transitions of P_i be ${}^\bullet P_i^{fire}$, the instantaneous firing speed of T_j^{fire} at time t for each $T_j^{fire} \in {}^\bullet P_i^{fire}$ be $v_{T_j^{fire}}(t)$, and the delay time of T_j^{fire} be $d_{T_j^{fire}}$. Thus the rate of tokens incoming to P_i is

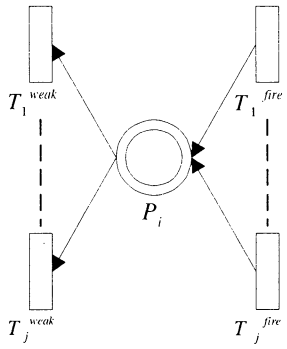


Figure 5: Transitions with a single input C place without nonreserved tokens

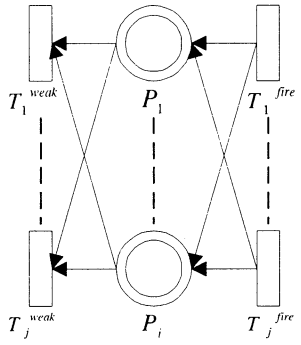


Figure 6: Transitions with more than one input C places without nonreserved tokens

represented by

$$v_{P_i}^{out}(t) = \sum_{T_j^{weak} \in T_{P_i}^{weak}} v_j^{weak}(t) \cdot I(P_i, T_j^{weak}) \quad (3)$$

and the rate of tokens outgoing from P_i is represented by

$$v_{P_i}^{out}(t) = \sum_{T_j^{weak} \in T_{P_i}^{weak}} v_j^{weak}(t) \cdot I(P_i, T_j^{weak}) \quad (4)$$

Hence, the instantaneous firing rate of T_j^{weak} with a single input C place without nonreserved tokens is given by

$$v_j^{weak}(t) = \min(V_j^{weak}, \frac{I(P_i, T_j^{weak})}{\sum_{T_j^{weak} \in T_{P_i}^{weak}} I(P_i, T_j^{weak})} v_{P_i}^{in}(t)) \quad (5)$$

Next, we formulate the instantaneous firing speeds with more than one input C places without nonreserved tokens. The speed of tokens $v_{P_i}^{out}(t)$ outgoing from P_i should be equal to or less than the speed of tokens $v_{P_i}^{in}(t)$ incoming to P_i. Moreover, instantaneous firing speeds of weakly (or strongly) enabled transitions always should be nonnegative and be equal to or less than the maximum speeds. Thus the calculation of instantaneous firing speeds implies an optimization problem to maximize the firing speeds under such constraints. Hence, the instantaneous firing speeds are determined using LP as follows:

$$\begin{aligned} &minimize\ V = \sum_j \alpha_j (V_j^{weak} - v_j^{weak}(t)) \\ &subject\ to: \\ &v_{P_i}^{in}(t) \geq v_{P_i}^{out}(t), \forall P_i; \\ &and\ \ V_j^{weak} \geq v_j^{weak}(t) \geq 0, \forall T_j^{weak} \in T_{P_i}^{weak} \end{aligned} \quad (6)$$

where coefficient α_j represents a priority of weakly enabled transition T_j^{weak}. The bigger the value of α_j is, the higher the priority of firing of T_j^{weak} is.

5 Conclusion

We have proposed new firing rules introducing the concept of delay time in order to represent continuous dynamics in the directed circuits consisting of C transitions and C places without tokens. Next, we have introduced a resolution policy that we use as many tokens as possible for firing subject to preempted enabling transitions, formulated by a LP problem. It is noted that LP becomes integer programming for a DPN due to integer markings, conflict resolution for a hybrid PN is formulated by mixed integer programming, and thus both the proposed firing rules and conflict resolution method are applicable for hybrid PNs. Control problems for the proposed model are future works.

References

[1] R. David and H. Alla, "Continuous Petri nets," *Proc. of 8th European Workshop on Application and Theory of Petri Nets*, Zaragoza, Spain, June, pp.275–294, 1987.

[2] T. Murata, "Petri nets: properties, analysis and applications," *Proc. of the IEEE*, Vol.77, No.4, April, pp.541–580, 1989.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Performance Improvement in an Internal Model Control of Discrete-Event Systems Based on Max-Algebra

Shiro Masuda
Dept. of Systems Eng.
Okayama Univ.
700-8530, JAPAN

Hiromitsu Suzuki
Fuchu Works
Toshiba Co., Ltd.
183-8511, JAPAN

Akira Inoue
Dept. of Systems Eng.
Okayama Univ.
700-8530, JAPAN

Yoichi Hirashima
Dept. of Systems Eng.
Okayama Univ.
700-8530, JAPAN

Abstract

In this paper, we consider an internal model control structure for discrete event systems based on max-algebra. Since a simple prediction method using the states of the process model for the derivation of the control law is used in the conventional method, the performance often shows undesirable one due to a certain disturbance and/or model-plant-mismatch. Therefore, we propose a new prediction strategy with a filter and a new control law using states of the controlled plant. Finally, we illustrate simulation results in order to make sure of the effectiveness of the proposed method.

1 Introduction

Max-algebra approach which is originally introduced by Cohen et al [1, 2] is a powerful design and performance evaluation tool of discrete-event systems (DES)[3]. Especially, the deterministic timed-event graphs, which are a kind of DES, can be linearly described by using max-algebra, so performance evaluation of DES can be done straightforwardly.

Recently, an attractive design method which has an internal model structure (IMC) [4] was proposed by Boimond and Ferrier [5]. This approach has the advantage that the designed closed system is robust to some disturbances and/or model-plant-mismatch because the control action is determined so that output errors between the real process and its model should be reduced. However, the performance of the closed loop system often becomes worse by the IMC control when disturbances and/or model-plant-mismatch have time-variant property and there is input delay in the controlled system because there exists possibility that the control action for compensation of output errors might be inadequate for the present disturbances and/or model-plant-mismatch.

Therefore, in this paper, we propose a new design method of DES based on max-algebra. The proposed method has the same internal model control structure as the conventional method. However, while the simplest prediction method is used in the conventional method, a new prediction method with a disturbance filter is used in the proposed method. Since the time-variant property of disturbances can be weakened by using a disturbance filter, the performance degradation due to periodical disturbances can be avoided. Furthermore, the proposed design method is modified so that the states of real process can be used for the control input. Since the states of the real process have informations on the present disturbances, the performance improvement can be expected. In fact, the proposed method to use the states of the real process is shown to be robust to an instantaneous disturbance of the process model. The effectiveness of the proposed method is shown by the comparison study between the proposed method and conventional method for an example of the two machines production system.

2 Max Algebra

A set $\boldsymbol{R} \cup \{\pm\infty\}$ supplied with two operations denoted by $\oplus$ and $\otimes$, where $\boldsymbol{R}$ is a set of real number, is called the *max-algebra* if the operators are defined in the following way.

$$x_i \oplus x_j \equiv \max(x_i, x_j) \quad (1)$$

$$x_i \otimes a_{ij} \equiv x_i + a_{ij} \quad (2)$$

The operators $\oplus$ and $\otimes$ are called "sum" or "addition", and "product" or "multiplication" respectively. Then the zero element ε and the one element e in max-algebra are

$$\varepsilon = -\infty, \quad e = 0 \quad (3)$$

The operation $\oplus$ and $\otimes$ between two matrices whose elements belong to $\boldsymbol{R} \cup \{\pm\infty\}$ are defined as follows:

$$[\boldsymbol{A} \oplus \boldsymbol{B}]_{ij} = \boldsymbol{A}_{ij} \oplus \boldsymbol{B}_{ij}, \quad (4)$$

$$[\boldsymbol{A} \otimes \boldsymbol{B}]_{ij} = \bigoplus_{k=1}^{n} \boldsymbol{A}_{ik} \otimes \boldsymbol{B}_{kj} \quad (5)$$

where $[\cdot]_{ij}$ stands for the element of i-th row and j-th column. $\boldsymbol{A}$ and $\boldsymbol{B}$ have appropriate dimensions with entries in $\boldsymbol{R} \cup \{\pm\infty\}$, and n is the number of column of $\boldsymbol{A}$ and the number of row of $\boldsymbol{B}$. Henceforth, the sign $\otimes$ is omitted as in usual linear algebra.

3 Internal Model Control Based on Max-Algebra (IMC-MA)

3.1 Problem Statements

Consider a single-input, single-output process described based on max-algebra given in the next:

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) \oplus \boldsymbol{b}u(k+1) \\ y(k) &= \boldsymbol{c}^{\mathrm{T}}\boldsymbol{x}(k) \end{aligned} \quad (6)$$

where $\boldsymbol{x}(k) \in \boldsymbol{R}^n$ is a state vector, and the each element $x_i(k)$ represents the k times occurrence time of the event i. $y(k)$ is the scalar output of the process which represents the k times occurrence time of the controlled event. $u(k+1)$ is the scalar input of the process which represents the $k+1$ times occurrence time of the accessible event. $\boldsymbol{A}$, $\boldsymbol{b}$ and $\boldsymbol{c}^{\mathrm{T}}$ have appropriate dimensions with eintries in $\boldsymbol{R} \cup \pm\infty$.

Let the reference signal be denoted $c(k)$. Then, the control objective is for the process output $y(k)$ to follow the reference signal $c(k)$ which means desired occurrence time sequences.

3.2 The Basic Structure of IMC-MA

The IMC method whose structure given in Fig.1 is effective for keeping the process output close to the reference input in case of mismatch between the process and its model. In this method, the controller calculates the control input which generates the given process output. Namely, it works as an inverse system of the process model. Hence, the desired control input can be obtained for the given reference signal. In addition, because the process model is included in the control system, the difference between the process output and the reference input due to disturbances and/or model-plant-mismatch can be easily detected and the reference signal can be modified by using the detected difference.

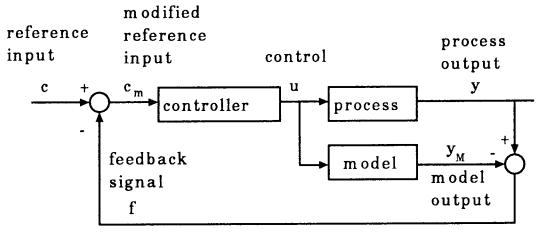


Figure 1: Basic internal model control structure

Now, we will show the control law using IMC structure in detail proposed in [5]. Let the process model be given as:

$$\begin{aligned} \boldsymbol{x}_M(k+1) &= \boldsymbol{A}_M\boldsymbol{x}_M(k) \oplus \boldsymbol{b}u(k+1) \\ y_M(k) &= \boldsymbol{c}_M^{\mathrm{T}}\boldsymbol{x}_M(k) \end{aligned} \quad (7)$$

where $\boldsymbol{x}_M(k)$, $y(k)$ and $u(k)$ are defined in the same way as (6). Then, using the modified reference signal $c_m(k)$ defined as:

$$c_m(k) = c(k) - f(k), \quad f(k) \stackrel{\triangle}{=} y(k) - y_M(k) \quad (8)$$

we can get $y(k) - c(k) = y_M(k) - c_m(k)$. Hence, the control objective $y(k) - c(k) = 0$ can be achieved by using the control input which generates the process model output $y_M(k)$ so that $y_M(k) - c_m(k) = 0$. Therefore, the next control law was proposed using IMC structure.

$$u(k+1) = -\Delta\{c_m(k+\delta+1) \oplus \Gamma\boldsymbol{x}_M(k)\} \quad (9)$$

where δ is characteristic number, and the smallest integer such that $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{A}^{\delta}\boldsymbol{b} \neq \varepsilon$. Δ and Γ is defined as:

$$\Gamma = \boldsymbol{c}^{\mathrm{T}}\boldsymbol{A}^{\delta+1}, \quad \Delta = \boldsymbol{c}^{\mathrm{T}}\boldsymbol{A}^{\delta}\boldsymbol{b} \quad (10)$$

It should be noted that the prediction of the modified reference signal $c_m(k+\delta+1)$ is needed for the calculation of the control input (9). From (8) that means the feedback signal $f(k)$ should be predicted. In the conventional method [5] the simplest prediction $f_p(k)$ of the feedback signal is used as shown in the next.

$$f_p(k) = f(k-q) \quad (11)$$

where it is assumed that q step past feedback signal can be accessible.

3.3 Performance Improvement in IMC-MA

As is shown in the previous subsection, the robustness to the disturbances and/or model-plant-mismatch can be achieved by using the conventional IMC-MA

method. However, there exists possibility that performance degradation might occur due to a certain disturbances. As such disturbances, a periodical disturbance and an instantaneous disturbance can be considered. If the system has delay between input and output signals, and disturbances have time-variant property, the reference input might be modified so that the output error becomes large in the conventional method because disturbances have changed when the output error is compensated.

Therefore, in this paper, the two methods for performance improvement in the presence of such disturbances are proposed. First, the method using a disturbance filter given as:

$$f_p(k) = \frac{(1-a)^q z^n}{(z-a)^q} f(k-q), \quad 0 \leq a < 1 \qquad (12)$$

is proposed, where z denotes the one step ahead operator, that is $zf(k) = f(k+1)$. Since the time-variant property of disturbances can be weakened by using the disturbance filter, the performance degradation due to periodical disturbances can be avoided. Secondly, the following control input using the states of the controlled process $\boldsymbol{x}(k)$ instead of the states of the process model $\boldsymbol{x}_M(k)$ is proposed as shown in the next.

$$u(k+1) = -\Delta \{c_m(k+\delta+1) \oplus \Gamma \boldsymbol{x}(k)\} \qquad (13)$$

Because the states of the controlled process $\boldsymbol{x}(k)$ have informations on the disturbances at the present step k, it is expected that the performance degradation can be avoided. In fact, the transient performance for an instantaneous disturbance can be improved in the proposed method.

Remark 3.1 *In order to realize the control law (13), the controlled system should satisfy the following condition.*

$$u(k+1) > \max_i x_i(k) \qquad (14)$$

4 Simulation Results

In this section, we consider a production system with two machines [5] as shown in Figure 2. A part which is given to the input stock is manufactured by machine 1 and machine 2 sequentially. In the production system, the machine 1 (the machine 2) does not load a new part until a new part has arrived at the input stock of machine 1 (the machine 2) and the machine 1 (the machine 2) has finished manufacturing the previous loaded part. Let the production system be described using the max-algebra. The holding time

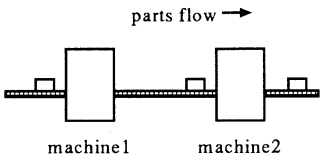


Figure 2: Production system

for manufacturing of the machine 1 and the machine 2 is denoted d_{M1} and d_{M2} respectively. $x_1(k)$, $x_2(k)$ and $x_3(k)$ are defined as the k-th starting time of the machine 1, the k-th ending time of the machine 1, and the k-th starting time of the machine 2. $y(k)$ is defined as the k-th ending time of the machine 2. $u(k)$ is defined as the k-th arriving time of a new part to the input port of the machine 1. Using these definitions, the production system can be described as:

$$\begin{cases} \boldsymbol{x}(k+1) = \begin{bmatrix} \varepsilon & e & \varepsilon \\ \varepsilon & d_{M_1} & \varepsilon \\ \varepsilon & d_{M_1} & d_{M_2} \end{bmatrix} \otimes \boldsymbol{x}(k) \\ \qquad\qquad \oplus \begin{bmatrix} e \\ d_{M_1} \\ d_{M_1} \end{bmatrix} \otimes u(k+1) \\ y(k) = \begin{bmatrix} \varepsilon & \varepsilon & d_{M_2} \end{bmatrix} \otimes \boldsymbol{x}(k) \end{cases} \qquad (15)$$

where $\boldsymbol{x}(k)^{\mathrm{T}} = (x_1(k), x_2(k), x_3(k))$. The control objective is for the output $y(k)$ to follow the desired ending time of the machine 2. Since $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{b} = d_{M1}d_{M2} \neq \varepsilon$, the characteristic number δ is zero. The conventional control input is given as:

$$u(k+1) = -\Delta\{c_m(k+1) \oplus \Gamma \boldsymbol{x}_M(k)\} \qquad (16)$$

At first, the next disturbances are assumed to be added to the controlled system.

- **periodical disturbandes** After 8th steps, the disturbances whose amplitude is 0.1 and the period is 2 steps are added to the machine 2.

Figure 3 and Figure 4 show the simulation result using the conventional control law and the proposed control law respectively in the case where the periodical disturbances are added. Here, the parameter a of a disturbance filter (12) is selected as $a = 0.9$ [1] . From these figures, we can see that the proposed method suppresses the output errors more than the conventional method.

The next, we consider the case where an instantaneous disturbance is added given in the next.

[1] In all figures the solid lines show the difference between the model output and the reference input, and the dotted lines show the difference between plant output and the reference input.

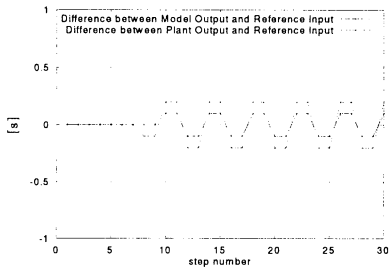


Figure 3: Simulation result using conventional prediction method in the case where a periodical disturbance is added

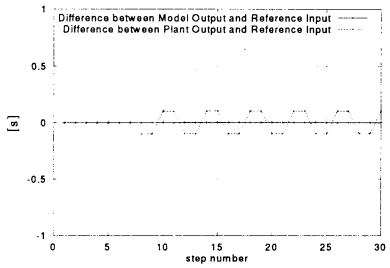


Figure 4: Simulation result using proposed prediction method in the case where a periodical disturbance is added

- instantaneous disturbance At the 10 steps an instantaneous disturbance is added on the machine 2.

Figure 5 and Figure 6 show the simulation result using the conventional control law and the proposed control law respectively in the case where the instantaneous disturbance are added. While an overshoot appears

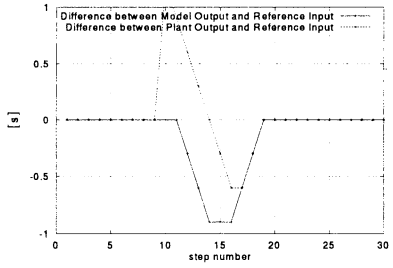


Figure 5: Simulation result using conventional control law in the case where an instantaneous disturbance is added

in the transient response of the conventional method, there is no overshoot in the transient response in the proposed method. Hence, we can see that the proposed method is more robust to an instantaneous disturbances than the conventional method.

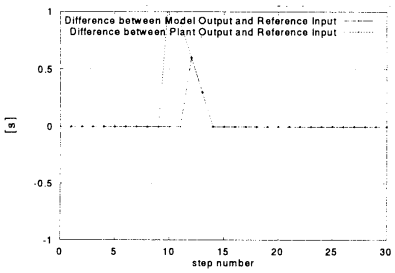


Figure 6: Simulation result using proposed control law in the case where an instantaneous disturbance is added

5 Conclusions

In this paper, we have proposed two methods for improving performance of a discrete-event systems based on max-algebra in the presence of a periodical disturbance or an instantaneous disturbance. The effectiveness of the proposed method has been shown by using numerical examples. The extension to the multivariable case remains as a future work.

References

[1] F. Baccelli, G. Cohen, G. J. Olsder and J. P. Quadrat, *Synchronization and Linearity - An Algebra for Discrete Event Systems*, Jhon Wiley & Sons, 1992.

[2] G. Cohen, P. Moller, J. P. Quadrat and M. Viot, "Algebraic tools for the performance evaluation of discrete event systems", *Proceedings of the IEEE*, Vol. 77, No.1, pp. 39-58, 1989.

[3] C. G. Cassandras, S. Lafortune and G. J. Olsder, "Introduction to the modelling, control and optimization of discrete event systems", *Trends in Control: A European Perspective* (Alberto Isidori, Ed.), Springer-Verlag, pp. 217-292, 1995.

[4] M. Morari and E. Zafiriou, *Robust Process Control*, Englewood Cliffs, NJ: Prentice-Hall, 1996.

[5] J. L. Boimond and J. L. Ferrier, "Internal model control and max-algebra: Controller design", *IEEE Trans. on Automatic Control*, Vol.AC-41, No. 3, pp. 457-461, 1996.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Autonomous Distributed Control of Manufacturing System by Scene Transition Nets

Seiichi Kawata
Department of Precision Engineering
Tokyo Metropolitan University
1-1 Minami-ohsawa, Hachioji-shi
Tokyo 192-0397 JAPAN

Mikio Haruyama
Sagamihara Machinery Works,
Mitsubishi Heavy Industries, LTD.
3000 Tana Sagamihara-shi
Kanagawa-ken, 229-1193 JAPAN

Abstract

In this paper, a discrete/continuous hybrid systems modeling combined with autonomous distributed control algorithms provides a new control method for manufacturing system in flexible manner. Actors in the STN are used to represent agents. And a communication field is defined in the STN. As an illustrative example, a manufacturing system simulation model is developed. Simulation results show that the proposed system can handle to represent autonomous distributed manufacturing systems.

Keywords

discrete/continuous hybrid systems, manufacturing systems, autonomous distributed systems, and scene transition nets

1 Introduction

As a future manufacturing system, many researchers try to find the new architecture of manufacturing system, which is capable of handling the rapid change of customer demands. As one of the expected solution to these problems, a concept of an autonomous and distributed manufacturing system is presented [1].

In order to develop the efficient manufacturing system, well described manufacturing model is needed. Most of the conventional manufacturing models are represented by discrete event systems model such as petri nets. However there are many continuous processes in real manufacturing system. Typical examples are in a chemical process and steel industry process that are described by sets of differential equations.

In this paper, a discrete/continuous hybrid systems modeling combined with autonomous distributed control algorithms [2] provides a new control method for manufacturing system.

In the first section of this paper, Scene Transition Nets are explained. The scene transition net (STN) is a new graphical representation to describe a discrete/continuous hybrid system, which uses the petri-net-like graph and formalism [3], [4] and [5]. The STN consists of scene, actor and transition. These scene, actor and transition respectively correspond to place, token and transition in petri net.

The scene transition net has the following characteristics.

1) The performer for each scene is identified; this information is lost in a petri net.
2) A system described in the form of a scene transition net can be simulated directly by an object-oriented program.
3) Appropriate class partitioning is possible by writing a scene transition net.
4) Appropriate scene partitioning is possible by writing a scene transition net.

The main elements of a scene transition net are as follows.
1) Scene box: Corresponds to a place in a petri net; denotes a scene, with a notation giving the performer.
2) Transition: Denotes the scene transition boundary; that is, denotes an event in the same way as a transition in a petri net.
3) Arc: Denotes the scene transition of an actor.

The firing rules for a scene transition net (that is, the scene transition boundaries) must be clearly stated for each transition. This is because firing rules that depend only on discrete variables, as in a petri net (the presence of a token on a place), do not suffice, because in many cases the firing conditions of a scene transition net depend on continuous variables associated with each actor.

In the next, autonomous distributed control method by scene transition nets is proposed. The proposed system consists of actors as modules, in other words, autonomous manufacturing units. Each module has sufficient intelligence to make decisions autonomously about its self-functions and motions. Actors as modules are also capable of exchanging information with other actors within a network (communication) field. What is important is that this exchange of information takes place not simply through ordinary electrical cables, but through a module's knowledge and capability of environment recognition, which enables it to exchange information when and with whom it is required.

The proposed method consists of following items.
1) Actors as modules which has capabilities such as self-decision making and self-check
2) Communication filed which enables for actors to communicate each other and make cooperation on it

As an illustrative example, small manufacturing model which has six machines and five automated guided vehicle are modeled by scene transition nets and developed the autonomous distributed control system by

using proposed method. The simulation results show the effectiveness of the proposed method.

2 A brief introduction to STN

The scene transition net (STN) is a new graphical representation to describe a discrete/continuous hybrid system, which uses the petri-net-like graph and formalism [1], [2], [3]. The STN consists of scene, actor and transition. These scene, actor and transition respectively correspond to place, token and transition in petri net.

The scene transition net has the following characteristics.

1. The performer for each scene is identified; this information is lost in a petri net.
2. A system described in the form of a scene transition net can be simulated directly by an object-oriented program.
3. Appropriate class partitioning is possible by writing a scene transition net.
4. Appropriate scene partitioning is possible by writing a scene transition net.

The main elements of a scene transition net are as follows.

1. Scene box: Corresponds to a place in a petri net; denotes a scene, with a notation giving the performer.
2. Transition: Denotes the scene transition boundary; that is, denotes an event in the same way as a transition in a petri net.
3. Arc: Denotes the scene transition of an actor.

The firing rules for a scene transition net (that is, the scene transition boundaries) must be clearly stated for each transition. This is because firing rules that depend only on discrete variables, as in a petri net (the presence of a token on a place), do not suffice, because in many cases the firing conditions of a scene transition net depend on continuous variables associated with each actor.

The STN is formally defined as follows.

Definition 1 Scene Transition Net (STN)

STN = (A, S, T, F, IP) is a scene transition net.
where
$A = (a_1, a_2 \ldots a_m)$ is a finite set of actors, $S = (s_1, s_2 \ldots s_n)$ is a finite set of scenes,
$T = (t_1, t_2 \ldots t_k)$ is a finite set of transitions, $F \subseteq (S \times T) \cup (T \times S)$ is a finite set of arcs, and
IP: $S \rightarrow \{a\}$, where $a \in A$ is the set of initial performers.

Actors, scenes, and transitions are defined as follows.

Definition 2 Actor

$a = (X, U, Y, O, D, C)$ is an actor.
where
$X = (x_1, x_2 \ldots x_p)^T$ is a state vector, $U = (u_1, u_2 \ldots u_q)^T$ is an input vector,
O: $X \rightarrow Y$ is an output function, and D: $(X, U, \Delta t) \rightarrow X$ is an actor dynamics.

Definition 3 Scene

$s = (CAS, L)$ is a scene.
where
$CAS = \{c\}$ is a casting, c is actor class, and L is a link.

Definition 4 Transition

$t = (B, f)$ is a transition.
where
B is a scene transition boundary and f is a state transition law.

To model the real world system based on the concepts of actors and scenes, we need to do the following.

1. Define the actors.
2. Define the scenes.
3. Define the relationships between the actors and the scenes.

Here, the relationships between the actors and the scenes mean how each actor moves along what course through which scene. To describe this graphically, we proposed STN.

3 Representation of autonomous distributed manufacturing system by STN

3.1 Actor as an autonomous distributed manufacturing module

The basic elements in autonomous distributed manufacturing system are autonomous distributed modules and communication field. Each autonomous distributed module has characteristics such as self-check, self-decision-making, mutual communication and cooperation capability, and homogeneity. Communication filed consists of facilities, which enable each module to be able to communicate with other modules.

The candidates for autonomous distributed modules in existing manufacturing system are parts, semi-products and AGVs (Automated guided vehicles). Actors are used to describe these elements modeling by STN. So, in this paper, we adopt actor in STN as an autonomous distributed manufacturing module. The actor is defined as sub-class of actor-class in object-oriented approach. So, the actor inherently has characteristics such as distributed nature and homogeneity.

To make actors of autonomous distributed modules in existing manufacturing system, we will build the following characteristic in the actor description.

1. self-decision-making capability
2. self-check capability
3. cooperative characteristic
4. communication capability among actors
5. to define communication field

3.2 Definition of communication field

To enable each actor communicate each other under the communication field, we will define the communication filed as following.

Definition : Communication Field (CF)

Communication filed (CF) is the virtual field through where each actor changes information such as conditions and states of other actors.

CF has following functions.

- to display the states of the system
- to exchange the information among actors through CF
-

3.3 The functions of actors

The functions of actors in this system are as followings.

- Action: conveyance, processing and assembly
- Decision-making: to decide how to do next according to the information through CF
- Cooperation: To resolve the conflict between one actor and the other actors.
- Collecting the information: To get the information through CF

4 Simulation

4.1 Simulation model

Figure 2 shows the simulation model.

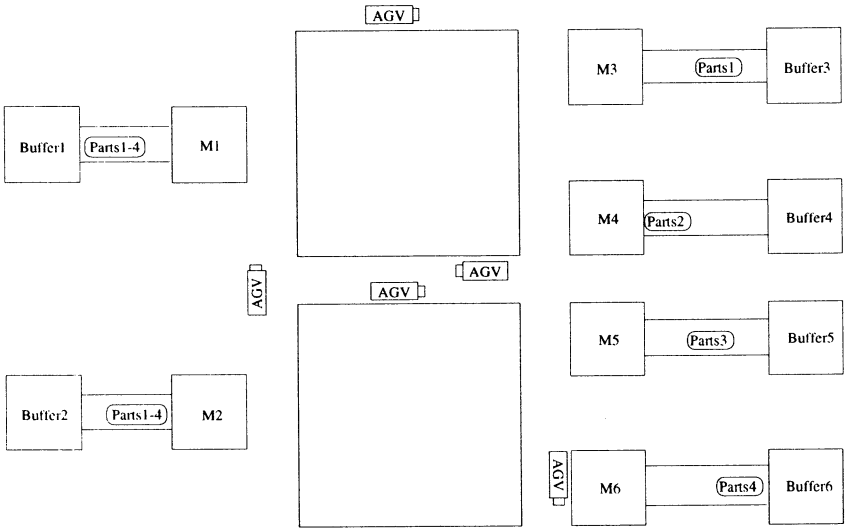


Figure 1 Simulation Model

The simulation model consists of six machines denoted by M1, M2, ... and M6. This manufacturing system manufactures four kinds of parts denoted by P1, P2, P3 and P4. In the first stage, M1 or M2 manufactures each part. The semi-products at exits of M1 and M2 are transferred to second stage by five AGVs (Automated guided vehicles) denoted by AGV1, AGV2, ... and AGV5. In the next stage, M3, M4, M5 and M6, in this order, manufacture these P1, P2, P3 and P4.

Rules of conveyance of AGV

1. Each AGV decide which AGV conveys the parts or semi-product from one machine to the other machine by cooperative negotiation.
2. AGV can convey one part or semi-product in one specific operation.
3. AGV can find the conveyance path by oneself
4. Each AGV has its own specific weight and battery system and in many cases this weight and/or battery system is different as shown in Table 1. So, each AGV has own specific speed.
5. Each part has own specific weight as shown in Table 2.

Table 1 Conditions of AGV

	Voltage of the battery[V]	Weight of the Vehicle[Kg]
AGV1	30	120
AGV2	25	120
AGV3	20	120
AGV4	25	100
AGV5	30	100

Table 2 Conditions of parts

	Weight of the Parts[Kg]
P1	120
P2	100
P3	80
P4	60

The algorithms for autonomous distributed control of AGVs are shown in Figure 2.

4.2 Simulation results

Table 3 shows the initial conveyance allocation for AGV and Table 4 shows the simulation results after malfunction of AGV. As can be seen from Table 4, malfunction occurred after allocating parts 6 to AGV1. Reallocation of AGV executed just after the malfunction. So, the system flexibly and autonomously fixed the malfunction problem.

Table 3 Initial allocation of AGV

Parts Number at M2	1	2	3	4	5	6	7	8
AGV Number	3	4	5	3	4	1	4	5

Table4 Reallocation of AGV

Parts Number at M2	1	2	3	4	5	6	7	8
AGV Number	3	4	5	3	4	1		
						2	5	4

5 Conclusion

An autonomous distributed manufacturing system control method described by scene transition nets has been developed in this paper and got the following results.

(1) The proposed system is based on discrete/continuos hybrid model.

(2) The proposed autonomous distributed control system uses actors in STN as autonomous distributed modules.

(3) Numerical examples show that the effectiveness of the proposed method.

References

[1] Edited by N. Okino, H. Tamura and S. Fujii, "Advances in Production Management Systems, perspectives and future challenges", selected, revised proceedings of the IFIP TC5/WG5.7 International Conference APMS'96, CHAPMAN & HALL, 1998

[2] M. Itoh, "Problems and future in study of autonomous distributed systems"(in Japanese). Journal of the SICE Vol.32, No.10, pp.789-796, 1993

[3] S. Kawata, S. Kawada, A. Watanabe, "New Simulation Model for Discrete-Continuous Hybrid Systems Using the Concept of Scene [Proposal for the Scene Transition Net (STN)]" (in Japanese), Transactions of the JSME, Series C, Vol.59, No.563, pp.1976-1982, 1993

[4] S. Kawada, S. Kawata, A. Watanabe, "The Formalism and Escapability of a Scene Transition"(in Japanese). Transactions of the JSME, Series C, Vol.62, No.594, pp.794-801, 1996

[5] S. Kawada, S. Kawata, A. Watanabe, "Some Extension of a Scene Transition Net by Introducing Actor Transformation and Multi-Aspect STN"(in Japanese), Transactions of the JSME, Series C, Vol.62, No.599, pp.2905-2912, 1996

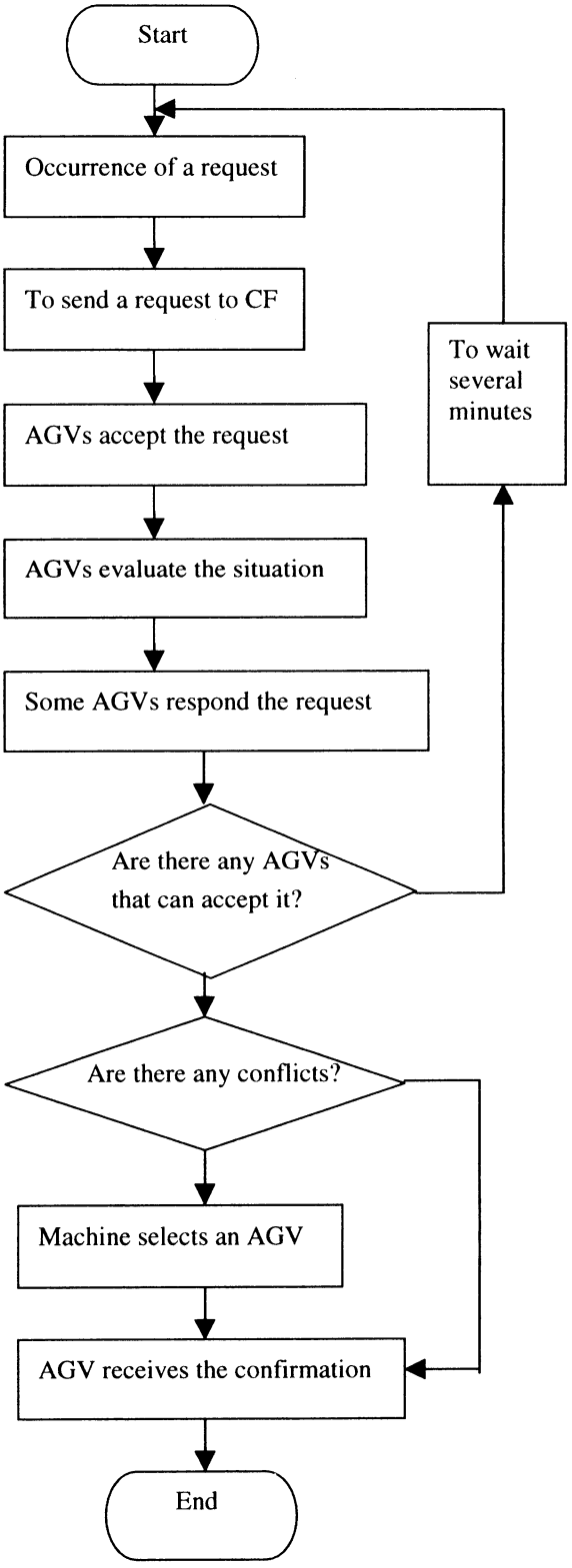


Figure 2 Algorithms for autonomous distributed control in AGV

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Introduction of Biochemical Allosteric Property for Creating a New Electronic Signal Transmission System under the Time Minimum Optimization.

H, Hirayama and *Y, Okita

Department of Public Health Asahikawa Medical College.
4-5 Nishi Kagura Asahikawa city 078. Japan.
* Department of Education Technology Natinal Institute of Special Education.
5-1-1. Nobi. Yokosuka city 239. Japan.

Abstract

We introduce an allosteric property of enzymes on biological membranes for creating a new electronic signal transmission system. In the allosteric enzymes, sequential bindings of substrates, activators and inhibitors induce molecular conformational changes in subunits of the inactive allosteric enzymes. We proposed the time minimum optimal control strategy for the allosteric reactions system as a dominating principle. The present mathematical model when it would be linked with the time optimal controller, will create a new signal transmission device.

Key words : Allosteric enzyme, Conformational change, Partial activation, Time optimal control.

1. Introduction

An allosteric enzyme[1,2] is composed of finite subunits that are configured symmetrically. Once a substrate has bound to one of these subunits, a conformational change occurs in a given region of an inactive stiff subunit. Then, the inactive enzyme becomes partially active and the subsequent substrates have easier access than the first substrate. The subsequent bindings of the substrate accelerate the molecular conformational change of the enzyme and the enzyme activity increases progressively. This can be expressed as an accelerated increase in the reaction rate in the successive partial activation of the enzyme.This allosteric property, when applied in electronic devices, produces a new signal transmission system.

In the present study, we propose an application of the biological allosteric property to signal transmission systems linked with the time minimum optimal control principle. We show that increasing the allosteric parameters accelerates the reactions and enhances the production significantly. The present method and modeling will be available for creating new electronic signal transmission devices.

2. Mathematical Modeling.

Fig 1 illustrates the sequential conformational changes of an allosteric enzyme composed of four identical subunits. The inactive form (Fig 1-a described by the filled squares) of the enzyme is markedly different from that of the active form (Fig 1-e described by the rounds). When a substrate **S** has bound to one of the four inactive subunits (filled square), conformational changes occur in two inactive subunits (Fig 1-b) and the enzyme is partially activated.The structure of the enzyme is modified so that subsequent substrates are more accessible to the partially activated enzyme (Fig 1-b) than to the first inactive form (Fig 1-a). Thereby, subsequent substrates are easier to bind to the residual three subunits than the first substrate is. The second substrate when it has bound to one of the three residual subunits (Fig 1-c) accelerates the conformational change of the other two subunits. The rate constants for the second, third and fourth substrates become significantly larger in a step by step manner than the rate constant for the first substrate. Finally the inactive form of the enzyme is converted to the active form that is bound to four substrates. This property of the allosteric enzyme is preserved even when an activator or an inhibitor has bound to the enzyme.

Fig 1. Sequential conformational changes in an allosteric enzyme.

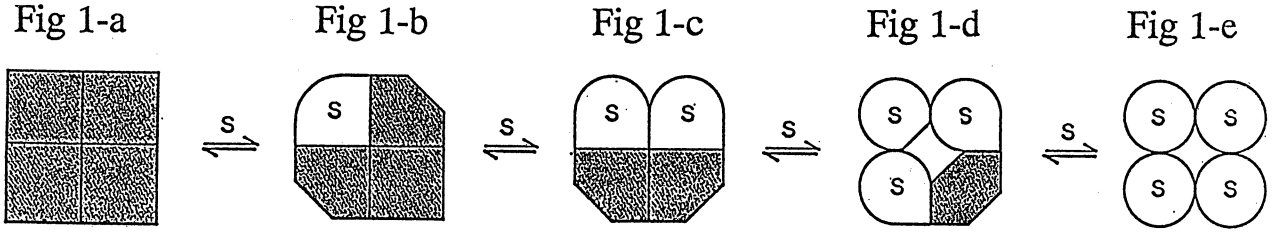
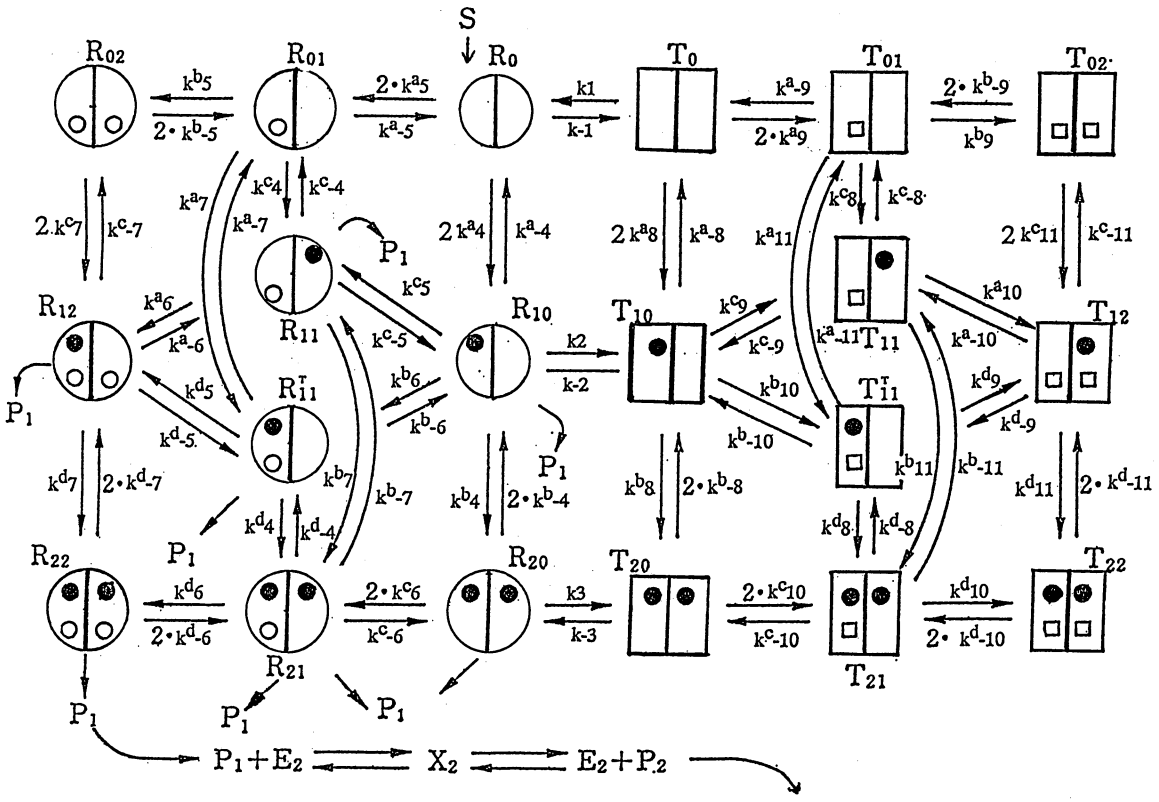


Fig 2. Allosteric reaction system with activators and inhibitors.



In the present investigation, we analyze two enzymes circuit on the biological membrane characterized by an allosteric property and a Michaelis Menten property (Fig 2). This reaction circuit is an extended version of Goldbeter's model [4] by Sakamoto (1988)[5]. The first step enzyme E1 is an allosteric enzyme having two allosteric subunits. The second step enzyme E2 is the Michaelis Menten type enzyme. The products operate as an activator and an inhibitor of the first step allosteric reaction circuit. Mutual conversion between the inactive and the active forms of the enzyme E1 occurs when they have bound only to the substrates. Active species (R) of the first step enzyme E1 system are illustrated by circles and inactive species (T) by squares. The active form of the first step enzyme has two subunits for substrates and another two different subunits for activators. The active form complexes (R_{10}, R_{11}, R^{T}_{11}, R_{12}, R_{22}, R_{21}, R_{20}) produce product P1. The inactive form of this enzyme also has two subunits for substrates and another two different subunits for inhibitors instead of the activators. The second step enzyme E2 reacts with P1 to produce product P2 through the substrate-enzyme e2 complex X2. P1 and P2 act as an activator and an inhibitor respectively on the E1 reaction system.

Following the experimental data [4] and modeling investigation[4,5], we set the following conditions.

1). Enzyme E1 can bind with a maximum of two substrates and two activators but not with any inhibitor when E1 is an active form. When the enzyme E1 is an inactive form, it can bind with a maximum of two inhibitors but not with any activator. Hence, there are two possible conformational changes of the enzyme E1 when it has bound with substrates, activators and inhibitors.

2). Substrate can bind to both of the active and the inactive forms of the enzyme E1 though the binding rate constants of the substrate to the active species are significantly larger than the rate constants to the inactive species.

3). Only the active form enzyme-substrate complexes can produce product P1.

4) An activator P1 binds exclusively to the active species. An inhibitor P2 binds only to the inactive species.

In Fig 2, a substrate is illustrated by a filled circle, an activator P1 by a circle and an inhibitor P2 by a square. The left side of the figure describes the conversions among the active species of the enzyme E1. The conversions in the vertical direction describe the substrate binding while those in the horizontal direction illustrate activator bindings (left R forms) and inhibitor bindings (right T forms). The conversions in the first row describes the species without substrate (R_{02}, R_{01}, R_{00}), in the second row describes the species that have bound with one substrate (R_{10}, R_{11}, R^{T}_{11}, R_{12}) and in the third row describes the species that have bound with two substrates (R_{20}, R_{21}, R_{22}). The conversions in the left most column express the substrate bindings to the two activators-enzyme complexes (R_{02}, R_{12} and R_{22}). The conversions in the second column (R_{01}--> (R_{11}, R^{T}_{11}) --> R_{21}) express the substrate binding to one activator-enzyme complexes. The conversions in the third column (R_{00}--> R_{10}-->R_{20}) express the substrate bindings to the enzyme E1 without an activator. In the figure, Rij indicates the active species bound with i substrates and j activators. Tij indicates the inactive species bound with i substrates and

j inhibitors. km is rate constant. In R^T_{11}, the substrate and the activator are positioned in parallel while symmetrical in R_{11}.

Allosteric properties of the enzyme E1 modifies the binding nature of the subsequent substrates to the enzyme-substrate complex[2,4,5].

1). For the species that have bound with **two activators** (R_{02}, R_{12} and R_{22}).

The rate constants of the substrate binding to these species are k^c_7, k^d_7. For the species that have bound with one activator (R_{01}, R_{11}, R^T_{11} and R_{21}), the rate constants of the substrate binding to these species are k^a_7, k^c_4 and k^d_4,k^b_7. For the species without activator (R_{00}, R_{10}, R_{20}), the rate constants of the substrate binding to these species are k^a_4, k^b_4.

2). For the species that have bound with **two substrates** (R_{22}, R_{21} and R_{20}).

The rate constants of an activator binding to these species are k^c_6, k^d_6. For the species that have bound with one substrate (R_{11}, R^T_{11} and R_{10}), the rate constants of the activator binding to these species are k^b_6, k^a_6 and k^c_5, k^d_5. For the species that are free of substrate (R_0, R_{01} and R_{02}), the rate constants of the activator binding to these species are k^a_5, k^b_5.

3). The rate constants of the substrate binding are larger in the complexes that have bound with the activator k^a_7, k^b_7, k^c_7, k^d_7 than those without the activator k^a_4, k^b_4.

We set the allosteric rate constants as following.

1. Substrate binding for R form species.

For the sequence of R_{02}----> R_{12}----> R_{22} (the species that have bound with two activators), we set

$$k7^d = m1 \; k7^c ,$$

for the sequence of R_{01}---> (R_{11}, R^T_{11}) -----> R_{21} (the species that have bound with one activator), we set

$$k7^b = m2 \; k4^c, \qquad k7^a = 1/m3 \; k4^d$$

and for the sequence of R_0---> R_{10} ---> R_{20} (the species having no activator) , we set

$$k4^b = m4 * k4^a$$

where m_n is an **allosteric parameter** and an integer.

2. Activator binding to R form species.

For the sequence of R_{20}----> R_{21}----> R_{22} (the species that have bound with two substrates), we set

$$k6^d = n4 \; k6^c ,$$

for the sequence of R_{10}---> (R_{11}, R^T_{11}) -----> R_{12} (the species that have bound with one substrate), we set

$$k6^a = n2 \; k5^c, \qquad k6^b = 1/n3 \; k5^d$$

and for the sequence of R_0---> R_{01} ---> R_{02} (the species having no substrate) , we set

$$k5^b = n1 \; k5^a$$

where n_m is an **allosteric parameter** and an integer.

3. Rate equations of the system.

Putting A as the concentration (mole) of the activator P1, I as the concentration (mole) of the inhibitor P2, we describe the transient changes of the concentrations of the species based on the mass action law.

$$\dot{R}_0 = -2k^a_4 S R_0 + k^a_{-4} R_{10} + k_1 T_0 - k_{-1} R_0 - 2k^a_5 A R_0 + k^a_{-5} R_{01}$$

$$\dot{R}_{10} = 2k^a_4 S R_0 - k^a_{-4} R_{10} - k^b_{14} R_{10} - k^b_4 S R_{10} + 2k^b_{-4} R_{20} - k^a_2 R_{10} + k^a_{-2} T_{10} - k^c_5 A R_{10} + k^c_{-5} R_{11} - k^b_6 R_{10} A + k^b_{-6} R^T_{11}$$

$$\dot{R}_{20} = k^b_4 S R_{10} - 2k^b_{-4} R_{20} - 2k_{14} R_{20} - 2k^c_6 A R_{20} + k^b_{-6} R_{21} - k_3 R_{20} + k_{-3} T_{20}$$

$$\dot{R}_{01} = -k^c_4 S R_{01} + k^c_{-4} R_{11} - k^a_7 S R_{01} + k^a_{-7} R^T_{11} - k^b_5 A R_{01} + 2k^b_{-5} R_{02} + 2k^a_5 A R_0 - k^a_{-5} R_{01}$$

$$\dot{R}_{11} = k^c_4 S R_{01} + k^c_{-4} R_{11} - k^a_{14} R_{11} - k^b_7 S R_{11} + k^b_{-7} R_{21} - k^a_6 A R_{11} + k^a_{-6} R_{12} + k^c_5 A R_{10} - k^c_{-5} R_{11}$$

$$\dot{R}^T_{11} = k^a_7 S R_{01} - k^a_{-7} R^T_{11} - k_{15} R^T_{11} - k^d_4 S R^T_{11} + k^d_{-4} R_{21} - k^d_5 A R^T_{11} + k^d_{-5} R_{12} + k^b_6 A R_{10} - k^b_{-6} R^T_{11}$$

$$\dot{R}_{21} = k^d_4 S R^T_{11} - k^d_{-4} R_{21} - k^c_{14} R_{21} + k^b_7 S R_{11} - k^b_{-7} R_{21} - k^c_{15} R_{21} - k^d_6 A R_{21} + 2k^d_{-6} R_{22} + 2k^c_6 A R_{20} - k^c_{-6} R_{21}$$

$$\dot{R}_{02} = k^b_5 A R_{01} - 2k^b_{-5} R_{02} - 2k^c_7 S R_{02} + k^c_{-7} R_{12}$$

$$\dot{R}_{12} = 2k^c_7 S R_{02} - k^c_{-7} R_{12} - k^a_{15} R_{12} + k^a_6 A R_{11} - k^a_{-6} R_{12} + k^d_5 A R^T_{11} - k^d_{-5} R_{12} - k^d_7 S R_{12} + 2k^d_{-7} R_{22}$$

$$\dot{R}_{22} = k^d_7 S R_{12} - 2k^d_{-7} R_{22} - 2k^b_{15} R_{22} + k^d_{-6} A R_{21} - 2k^d_{-6} R_{22}$$

$$\dot{T}_0 = -k_1 T_0 + k_{-1} R_0 - k^a_{-9} T_{01} - 2k^a_9 T_{10} I - 2k^a_8 S T_0 + k^a_{-8} T_{10}$$

$$\dot{T}_{10} = 2k^a_8 S T_0 - k^a_{-8} T_{10} - k^b_8 S T_{10} + 2k^b_{-8} T_{20} + k^a_2 R_{10} - k^a_{-2} T_{10} - k^c_9 I T_{10} + k^c_{-9} T_{11} - k^b_{10} I T_{10} + k^b_{-10} T^T_{11}$$

$$\dot{T}_{20} = k^b_8 S T_{10} - 2k^b_{-8} T_{20} + k_3 R_{20} - k_{-3} T_{20} - 2k^c_{10} I T_{20} + k^c_{-10} T_{21}$$

$$\dot{T}_{01} = -k^c_8 S T_{01} + k^c_{-8} T_{11} - k^a_{11} S T_{01} + k^a_{-11} T^T_{11} - k^c_{-9} T_{01} + 2k^a_9 I T_{10} + 2k^b_{-9} T_{02} - k^b_{-9} I T_{01}$$

$$\dot{T}_{11} = k^c_8 S T_{01} - k^c_{-8} T_{11} - k^b_{11} S T_{11} + k^b_{-11} T_{21} + k^c_9 I T_{10} - k^c_{-9} T_{11} - k^a_{10} I T_{11} + k^a_{-10} T_{12}$$

$$\dot{T}^T_{11} = k^a_{11} S T_{01} - k^a_{-11} T^T_{11} + k^b_{10} I T_{10} - k^b_{-10} T^T_{11} - k^d_9 I T^T_{11} + k^d_{-9} T_{12} - k^d_8 S T^T_{11} + k^d_{-8} T_{21}$$

$$\dot{T}_{21} = k^d_8 S T^T_{11} - k^d_{-8} T_{21} + k^b_{11} S T_{11} - k^b_{-11} T_{21} + 2k^c_{10} I T_{20} - k^c_{-10} T_{21} - k^d_{10} I T_{21} + 2k^d_{-10} T_{22}$$

$$\dot{T}_{02} = k^b_9 I T_{01} - 2k^b_{-9} T_{02} - 2k^c_{11} S T_{02} + k^c_{-11} T_{12}$$

$$\dot{T}_{12} = 2k^c_{11} S T_{02} - k^c_{-11} T_{12} + k^a_{10} I T_{11} - k^a_{-10} T_{12}$$

Fig 3. The temporal changes of the concentration (mM) of the activated species at the standard parameter values.

Fig 3-a

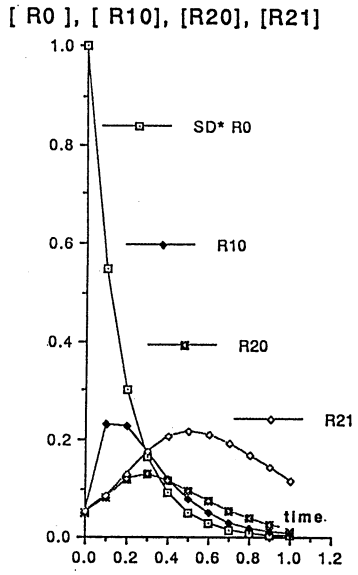


Fig 3-b

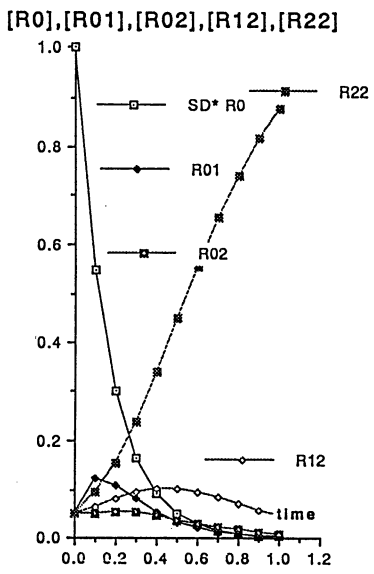
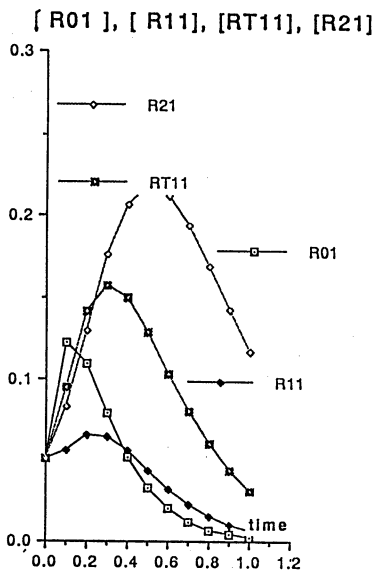


Fig 3-c



For the organization strategy of the present complicated reaction schema, we propose a time minimum optimal control strategy . This assumes that the system is organized so as to transmit the exogenous information as fast as possible. The performance function of the time minimum optimal control is simply

$$J(u) = \int_{t0}^{T} 1.0 \; dt = T$$

The time optimized differential equations for the co-state variable λn can be obtained by differentiating the Hamiltonian with respect to the corresponding state variables.

$$\frac{\partial \lambda n}{\partial t} = - \frac{\partial H}{\partial Xn}$$

where Xn are the state variables assigned for the concentration of the species for the present reaction

2. Results and Discussion

Fig 3-a shows the temporal concentration changes of the species composed of the sequential conversion from R_0 ---> R_{10} ---> R_{20} ---> R_{21}. Rapid decrease in R_0 and successive gradual increases in R_{10}, R_{20} and R_{21} describe the progress of the conversion of R_{00} to substrate-enzyme complexes. Fig 3-b shows the sequential conversion from R_0 ---> R_{01} ---> R_{02}---->R_{12} ---> R_{22} . With the progress of the reaction, the species are converted successively to R_{22}. Fig 3-c shows the the sequential conversion from R_{01} ---> (R_{11}, R^T_{11}) ---> R_{21} .

We have introduced a new concept "an allosteric property " of the biochemical enzyme[2,4,5] to an electronic signal transmission circuit. We could show that the allosteric effect had a marked influence on the species conversion process. With the increases in the allosteric parameters, concentrations of all the species changed more rapidly and in larger amounts than those at the standard allosteric parameter value. For the organization principle, we proposed a time optimal control strategy for the activation and inhibition processes of the allosteric reaction. This principle will also be available for evaluating the signal transmission function how the signal is transmittedefficiently . The present model can well describe the complicated sequential signal transmission process of the biochemical reaction by computing the temporal changes in the concentration of the species. As long as we analyze the linear and physiological range of the present system, however, the present method will be available for developing a signal transmission circuit.

3. Conclusion.

We proposed an application of the biological allosteric property for an electronic signal transmission system linked with the time minimum optimal control principle. The present method will be available for creating new electronic signal transmission devise.

4.References.

1. Colquhoun D, Sakmann B. (1983) In : Single channel recording. New York. Plenum.
2. Monod J, Wyman J, Changeux JP (1965). J. Molecular Biology 12 : 88-118
3. Jerne NK. (1974) Ann. Immunol 125: 373-389
4. Goldbeter A (1972) Biophysical .J 12 : 1302-1315.
5. Sakamoto N, Hayashi K. (1986) In Dynamic analysis of enzyme systems. Springer Verlag. Tokyo.
6. Prigogine I, Nicolis, G (1971) Quarterly Reviers of Biophysics 4 : 107-148
7.Glansdorff P, Prigogine I (1971) In Thermodynamics Theory of structure, stability and Fluctuation Wiley Iner Science NY.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Diffusively Coupled Multi-Neural Networks for Optimization Problems

Ryota Horie and Eitaro Aiyoshi
Department of Instrumentation Engineering, Keio University, Yokohama, Japan 223-8522
horie@sys.inst.keio.ac.jp

Abstract

We propose new mutually coupled neural networks named Diffusively Coupled Multi-Neural Networks(DCMNN) to solve global optimization problems. First, we propose a new multi-points dynamical searching model named Diffusively Coupled Gradient Systems (DCGS) based on the reaction-diffusion system. Second, we propose the DCMNN which realize the proposed DCGS with a braking operator for quadratic programming problems with $[0,1]$-interval constraints. Two types of structure named external type DCMNN (E-DCMNN) and internal type DCMNN (I-DCMNN) are proposed. In addition, we indicate that the proposed I-DCMNN can approximate dynamics of the time-dependent Ginzburg-Landau (TDGL) equation. We carried out computer simulations and confirm the ability of the proposed DCMNN.

1 Introduction

In the study of artificial neural networks (N.N.), the Hopfield type N.N. and its application to solving optimization problems have been the central topics [1]. The Hopfield type N.N. realizes a dynamical searching model for optimal points of quadratic or bilinear objective functions with the $[0,1]$-interval constraints[2]. The Hopfield type N.N. is a local searching model and its dynamics can not converge to global optimal points without suitable initial conditions.

To solve global optimization problems, multi-points searching models have been studied[3]. In these models, state transition of each searching point is determined by its own local information on a objective function and interaction with other searching points. Genetic algorithms can be regard as a kind of multi-points searching model.

In order to solve global optimization problems, we propose new mutually coupled N.N.s named Diffusively Coupled Multi-Neural Networks(DCMNN). In section 2, we propose a new multi-points dynamical searching model named Diffusively Coupled Gradient Systems (DCGS) based on the reaction-diffusion system. In section 3, we propose the DCMNN which realize the proposed DCGS with a braking operator for quadratic programming problems with $[0,1]$-interval constraints. Concretely, these are external type DCMNN (E-DCMNN) in which the outputs from neurons are coupled and internal type DCMNN (I-DCMNN) in which the inner variables of neurons are coupled. In section 4, we indicate additionally that the proposed I-DCMNN in which each N.N. is composed of one neuron can approximate dynamics of the time-dependent Ginzburg-Landau (TDGL) equation with a fourth order free energy. In section 5, we carried out computer simulations to confirm the ability of the proposed DCMNN.

2 Diffusively coupled gradient system

In order to solve optimization problems

$$\min_{\boldsymbol{x}} E(\boldsymbol{x}), \tag{1}$$

we consider reaction-diffusion systems[4, 5, 6] whose reaction term is a gradient system for a objective function $E(\boldsymbol{x})$

$$\frac{\partial \boldsymbol{x}_i(\boldsymbol{r};t)}{\partial t} = -\frac{\partial E(\boldsymbol{x}(\boldsymbol{r};t))}{\partial \boldsymbol{x}_i} + D\nabla^2 \boldsymbol{x}_i(\boldsymbol{r};t) \qquad i=1,\cdots,n, \tag{2}$$

where $\boldsymbol{x}\in\boldsymbol{R}^n$ is the state variable; $\boldsymbol{r}\in S\subset\boldsymbol{R}^m$ is the spatial variable; and $t\in[0,\infty)$ is the time variable. By discretization of the spatial variable $\boldsymbol{r}$ to a finite set Z composed of K points and Turing's first order approximation[7] for the diffusion term on these points $\boldsymbol{z}\in Z\subset\boldsymbol{Z}^m$, such as

$$\nabla^2 \boldsymbol{x}_i(\boldsymbol{r};t) \simeq \{<< \boldsymbol{x}_i(\boldsymbol{z};t) >> -\boldsymbol{x}_i(\boldsymbol{z};t)\}, \tag{3}$$

we have coupled ordinary differentail equations

$$\frac{d\boldsymbol{x}_i(\boldsymbol{z};t)}{dt} = -\frac{\partial E(\boldsymbol{x}(\boldsymbol{z};t))}{\partial \boldsymbol{x}_i}$$

$$+D\{<< x_i(z;t) >> - x_i(z;t)\}$$
$$i = 1, \cdots, n. \quad (4)$$

We regard the equations (4) as a multi-points dynamical searching model. Each of K gradient systems on the discrete points, as a local optimizer, searches minima of the objective function under diffusive interaction. We call the system (4) Diffusively Coupled Gradient Systems (DCGS).

For the structure of diffusive coupling, we consider following types:

- nearest-neighbor coupling on one dimensional lattice

$$<< x_i(z;t) >> = \frac{1}{2}\{x_i(z-1;t) + x_i(z+1;t)\} \quad (5)$$

with periodic boundary condition

- nearest-neighbor coupling on two dimensional square lattice

$$<< x_i(z_1, z_2;t) >>$$
$$= \frac{1}{4}\{x_i(z_1, z_2-1;t) + x_i(z_1-1, z_2;t)$$
$$+x_i(z_1+1, z_2;t) + x_i(z_1, z_2+1;t)\} \quad (6)$$

with periodic boundary conditions

- nearest and next nearest-neighbor coupling on two dimensional square lattice

$$<< x_i(z_1, z_2;t) >>$$
$$= \frac{1}{12}\{2x_i(z_1, z_2-1;t) + 2x_i(z_1-1, z_2;t)$$
$$+2x_i(z_1+1, z_2;t) + 2x_i(z_1, z_2+1;t)$$
$$+x_i(z_1-1, z_2-1;t) + x_i(z_1+1, z_2-1;t)$$
$$+x_i(z_1-1, z_2+1;t) + x_i(z_1+1, z_2+1;t)\} \quad (7)$$

with periodic boundary conditions

- globally coupling[8]

$$<< x_i(z;t) >> = \frac{1}{K}\sum_{z=1}^{K} x_i(z;t). \quad (8)$$

3 Diffusively coupled multi-neural networks

We propose Diffusively Coupled Multi-Neural Networks (DCMNN) which realize the proposed DCGS (4). For the realization by structures of the N.N., we consider quadratic programming problems with variables constrained on the closed interval $[0, 1]$'s

$$\min_{\boldsymbol{x}} E(\boldsymbol{x}) = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} w_{ij}x_i x_j + \sum_{i=1}^{n}\theta_i x_i$$
$$+\frac{1}{\tau}\sum_{i=1}^{n}\{x_i \ln x_i + (1-x_i)\ln(1-x_i)\} \quad (9a)$$
$$\text{subj.to} \quad 0 \leq x_i \leq 1, \quad i = 1, \cdots, n. \quad (9b)$$

The second term of the objective function (9a) is a barrier function which make gradients on boundaries of the constraints $[0, 1]^n$ infinity.

3.1 External type DCMNN

We consider the proposed DCGS (4) with a braking operator

$$\frac{dx_i(z;t)}{dt} = x_i(z;t)(1 - x_i(z;t))$$
$$\times\{-\frac{\partial E(\boldsymbol{x}(z;t))}{\partial x_i} + D\{<< x_i(z;t) >> - x_i(z;t)\}\}$$
$$i = 1, \cdots, n. \quad (10)$$

The function $x_i(1 - x_i)$ is a braking operator which reduces the gradient and the diffusive coupling term to zero as the variable x_i approaches the boundary of the $[0, 1]$-interval constraint.

We consider K N.N.s composed of n neurons with a nonlinear input-output function f, such as

$$x_i(z;t) = f(u_i(z;t)), \quad i = 1, \cdots, n, \quad (11)$$

where $u_i(z;t) \in R$ denotes a inner state variable of the i-th neuron in the N.N. located on z, and $x_i(z;t) \in [0, 1]$ denotes the output of the neuron. To compare the differentiation of (11) by the time variable t

$$\frac{dx_i(z;t)}{dt} = \frac{df(u_i(z;t))}{du_i}\frac{du_i(z;t)}{dt} \quad (12)$$

with the equation (10), we have

$$\frac{df(u_i(z;t))}{du_i} = x_i(z;t)(1 - x_i(z;t)) \quad (13a)$$
$$\frac{du_i(z;t)}{dt} = -\frac{\partial E(\boldsymbol{x}(z;t))}{\partial x_i}$$
$$+D\{<< x_i(z;t) >> - x_i(z;t)\}. \quad (13b)$$

To solve (13a), we use the well known sigmoid function for the input-output function of the neurons

$$f(u_i) = \frac{1}{1 + \exp(-u_i)}. \quad (14)$$

From (9a),(13b) and (14), we have a structure of a new mutually coupled N.N.s

$$\frac{du_i(\boldsymbol{z};t)}{dt} = -\frac{1}{\tau}u_i(\boldsymbol{z};t) - \left\{\sum_{j=1}^{n} w_{ij}x_j(\boldsymbol{z};t) + \theta_i\right\} + D\left\{<< x_i(\boldsymbol{z};t) >> - x_i(\boldsymbol{z};t)\right\} \quad i = 1, \cdots, n \qquad (15)$$

in which the outputs from neurons of each N.N. are coupled diffusively with other N.N.s' neurons. We call the prposed mutually coupled N.N.s External type DCMNN (E-DCMNN). For the structure of coupling among N.N.s, the nearest-neighbor coupling on one and two dimensional lattice (5),(6), the nearest and next nearest-neighbor coupling on two dimensional lattice (7), and the globally coupling (8) are considered. Note that the celluler neural networks[9] can be regarded as the proposed E-DCMNN in which each N.N. are composed of one neuron.

3.2 Internal type DCMNN

We consider the proposed DCGS with a braking operator (10) which have a nonlinear diffusive coupling term through a inverse function of the sigmoid function (14)

$$\frac{dx_i(\boldsymbol{z};t)}{dt} = x_i(\boldsymbol{z};t)(1 - x_i(\boldsymbol{z};t)) \times \left\{-\frac{\partial E(\boldsymbol{x}(\boldsymbol{z};t))}{\partial x_i} + D\{<< f^{-1}(x_i(\boldsymbol{z};t)) >> - f^{-1}(x_i(\boldsymbol{z};t))\}\right\} \quad i = 1, \cdots, n. \qquad (16)$$

In the same way of the proposed E-DCMNN (14),(15), to consider neurons (11) and compare their differentiation by t (12) with (16), we have the sigmoid function (14) for the input-output function of the neurons and a structure of a new mutually coupled N.N.s

$$\frac{du_i(\boldsymbol{z};t)}{dt} = -\frac{1}{\tau}u_i(\boldsymbol{z};t) - \left\{\sum_{j=1}^{n} w_{ij}x_j(\boldsymbol{z};t) + \theta_i\right\} + D\left\{<< u_i(\boldsymbol{z};t) >> - u_i(\boldsymbol{z};t)\right\} \quad i = 1, \cdots, n \qquad (17)$$

in which the inner variables of neurons of each N.N. are coupled diffusively with other N.N.s' neurons. We call the prposed mutually coupled N.N.s Internal type DCMNN (I-DCMNN).

4 Time-dependent Ginzburg-Landau equation

We indicate that the proposed I-DCMNN (14),(17) can approximate dynamics of the time-dependent Ginzburg-Landau (TDGL) equation. The TDGL equation has been studied as phenomenological models for nonlinear relaxation phenomena in the nonequilibrium statistical phisics[10] and has studied for a basic model to design decentralized autonomous systems[11]. We suppose the TDGL equation has a fourth order free energy, such as

$$E(\psi) = f_0 - h\psi - \frac{a}{2}\psi^2 + \frac{b}{4}\psi^4. \qquad (18)$$

To use Turing's first order approximation for the diffusion term, we have

$$\frac{\partial\psi(\boldsymbol{z};t)}{\partial t} = (h + a\psi(\boldsymbol{z};t) - b\psi(\boldsymbol{z};t)^3) + D\left\{<< \psi(\boldsymbol{z};t) >> - \psi(\boldsymbol{z};t)\right\}. \qquad (19)$$

We consider the proposed I-DCMNN (14),(17) in which each N.N. is composed of one neuron

$$\begin{aligned}\frac{\partial u(\boldsymbol{z};t)}{\partial t} &= -\frac{1}{\tau}u(\boldsymbol{z};t) - \{wx(\boldsymbol{z};t) + \theta\} \\ &\quad + D\left\{<< u(\boldsymbol{z};t) >> - u(\boldsymbol{z};t)\right\} \\ &= -\frac{1}{\tau}u(\boldsymbol{z};t) - wf(u(\boldsymbol{z};t)) - \theta \\ &\quad + D\left\{<< u(\boldsymbol{z};t) >> - u(\boldsymbol{z};t)\right\}. \qquad (20)\end{aligned}$$

To expand the sigmoid function f (14) by the Maclaurin's formula to order three, we have

$$f(u) \simeq \sum_{k=0}^{3} \frac{1}{k!}\frac{d^k f(u(0))}{du^k} \qquad (21a)$$

$$f(0) = 0.5 \qquad (21b)$$

$$\frac{df(u(0))}{du} = x(0) - x(0)^2 = 0.25 \qquad (21c)$$

$$\frac{d^2 f(u(0))}{du^2} = x(0) - 3x(0)^2 + 2x(0)^3 = 0.0 \qquad (21d)$$

$$\frac{d^3 f(u(0))}{du^3} = x(0) - 7x(0)^2 + 12x(0)^3 - 6x(0)^4 = 0.125. \qquad (21e)$$

To compare (19) with (20) and (21), we have coefficients of the proposed I-DCMNN (20) to approximate dynamics of the TDGL equation (19), such as

$$\tau = -\frac{1}{a - 12b} \qquad (22a)$$

$$w = -\frac{6}{0.125}b \qquad (22b)$$

$$\theta = -h + 24b. \qquad (22c)$$

We emphasize that the proposed I-DCMNN (20) can be used as a device realizing the TDGL equation (19).

5 Computer simulation

We carried out computer simulation for simple quadratic or bilinear problems with the $[0,1]$-interval constraints (9) to confirm the searching ability of the proposed DCMNN. We compare the proposed E-DCMNN (14),(15) and the proposed I-DCMNN (14),(17), and investigate the effects of the diffusive coefficient, the structure of coupling and the number of N.N. composing the DCMNN. From the simulation results, we confirm that the proposed DCMNN has following features:

1. From initial points with each of which the single N.N. is tarpped in local minima, the proposed DCMNN can converge to a global optimum by effect of many-bodies interaction with suitable diffusive coeficients.

2. With the same diffusive coeficients, the proposed I-DCMNN converge more quickly than the proposed E-DCMNN.

3. From initial points generated randomly, the convergence ratios of the proposed I-DCMNN to a global optimum are higher than that of the single N.N. when the I-DCMNN is composed of a few N.N.s, and the structure of coupling is one dimensional lattice (5) or globally coupling (8) with small diffusive coefficients.

4. The proposed I-DCMNN (14),(20),(22) realizing the TDGL equation (19) on two dimensional square lattice has phase-ordering dynamics like the spinodal decomposition which the TDGL equation describes[10].

6 Conclusion

We propose new mutually coupled neural networks named Diffusively Coupled Multi-Neural Networks (DCMNN) which realize dynamical searching model named Diffusively Coupled Gradient System (DCGS) for quadratic programming problems with $[0,1]$-interval constraints. We propose two types of structure named external type DCMNN (E-DCMNN) and internal type DCMNN (I-DCMNN). The proposed DCMNN have ability to search global optimal points by effect of many-bodies interaction. The proposed I-DCMNN can approximate dynamics of the time-dependent Ginzburg-Landau equation.

In the future research, we will study the DCMNN realizing reaction-diffusion systems whose reaction terms are not gradient system.

References

[1] J. J. Hopfield and D. W. Tank, ""Neural" Computation of Decision in Optimization Problems", *Biol. Cybern.* , Vol. 52, pp.141-152, 1985.

[2] E. Aiyoshi and A. Yoshikawa, "Optimization by Neural Networks: Their Questions and Subjects – Another Approach–"(in Japanese), *J. SICE*, Vol. 34, No. 5, pp.358-366, 1995.

[3] H. Sugata, T. HAGINO, K. SHIMIZU, "Global Optimization Method Applying Chaos in Multi-Trajectory Inertial System"(in Japanese), *Trans. IEICE A*, Vol. J79-A, No. 7, pp.1-8, 1996

[4] R. Kapral and K. Showalter(Eds.), *Chemical Waves and Patterns*, Kluwer Academic Publishers, 1995.

[5] E. Ben-Jacob,O. Shochet,A. Tenenbaum, I. Cho -en, A. Csirók and T. Vicsek, "Complex bacterial patterns", *Nature*, Vol. 373, pp.566-567, 1995.

[6] K. Yoshikawa, I. Motoike and K. Kajiya, "Design of Excitable Field Towards a Novel Parallel Computation", *IEICE Trans. Electron.*, Vol. E80-C, No. 7, pp.931-934, 1997.

[7] A. M. Turing, "The Chemical Basis of Morphogenesis", *Philos. Trans. R. Soc. Lond.*, Vol. B237, pp.37-72, 1952.

[8] K. Kaneko, "Globally coupled circle maps", *Phisica D* , Vol. 54, pp.5-19, 1991.

[9] L. O. Chua and L. Yang, "Cellular Neural Networks :Theory", *IEEE Trans. Circuits and Systems*, Vol. CAS-35, pp.1257-1272, 1988.

[10] Y. Oono and S. Puri, "Study of Phase-Separation Dynamics by Use of Cell Dynamical Systems. I. Modeling", *Phis. Rev. A*, Vol. 38, No. 1, pp.434-453, 1988.

[11] H. Yuasa, S. Ito and M. Ito, "Associative Memory with the Reaction-Diffusion Equation", *Biol. Cybern.*, Vol. 76, pp.129-137, 1997.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Learning Algorithm for Intermodular Connection of Multimodular Associative Networks

Masayuki Mochizuki† and Haruyuki Minamitani†
Email:mochi@appi.keio.ac.jp

†Department of Applied Physics and Physico-Informatics, Faculty of Science and Technology, Keio University,
3-14-1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa 223-8522, Japan

ABSTRACT

In this study, we propose an intermodular connection network ξ-connection and an algorithm for multimodular associative networks. This network is characterized by its effectiveness in learning exclusively the connection among the newly added modules and the directly connected modules. And our proposed learning algorithm aims at achieving a high degree of noise tolerance in recollection capability by examining the salient differences among learning instances and employing linear coordinate transformation. Furthermore, as learning is made localized between two directly connected modules, it is particularly appropriate for learning with incremental addition of modules. Due to the fact that only the connections among the newly added module and the directly connected ones are considered, the size of the network is expected not to grow rapidly.

Keyword:
Algorithms, Learning, Memory, Multimodule,
Associative Network

1. Introduction

Neural networks have been applied to motor control, character recognition and etc., and process various problems at high speed. Up to now, many network models and learning algorithms have been proposed.

However, as the dealt problems is complex, size of networks become larger and learning become more difficult. Recently, modular technique have been proposed to reduce computational complexity in learning[1,2,4]. One of network models to which modular technique was applied, is multimodular associative networks[4].

When information attributes are added, it is necessary to add modules. However, most multimodular associative networks lack the appropriate structure that could adaptively accomodate newly added modules to the existing ones. In the multimodular associative network MuNet proposed by Ohsumi et al., the intermodular connections between all modules have to be relearned and adjusted when a new module is added, which involves a high cost in computation time.

In this study, We present an intermodular connection network which is characterized by its effectiveness in learning exclusively the connections among the newly added modules and the directly connected modules, and its high degree of noise tolerance in recollection capability.

2. Multimodular Associative Network MuNet

In this section multimodular associative network MuNet is described.

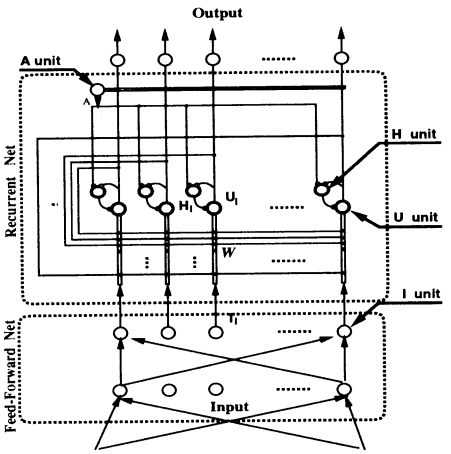


Figure 1: Structure of MuNet's module

Multimodular associative network MuNet[4], which can do difficult memories and associations such as many-to-many association, was proposed by Ohsumi et al.

Each module of MuNet is a hybrid network which is composed of recurrent network for auto-association and feed-forward network to learn interdependency among modules (Figure 1).

The feed-forward network recalls the superposed pattern of all patterns corresponding to input patterns (output

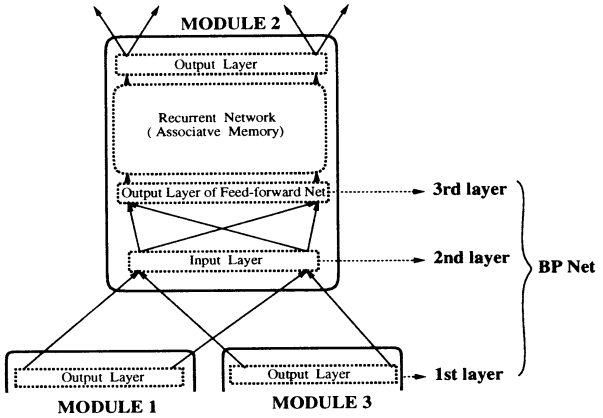


Figure 2: A intermodular connection of MuNet

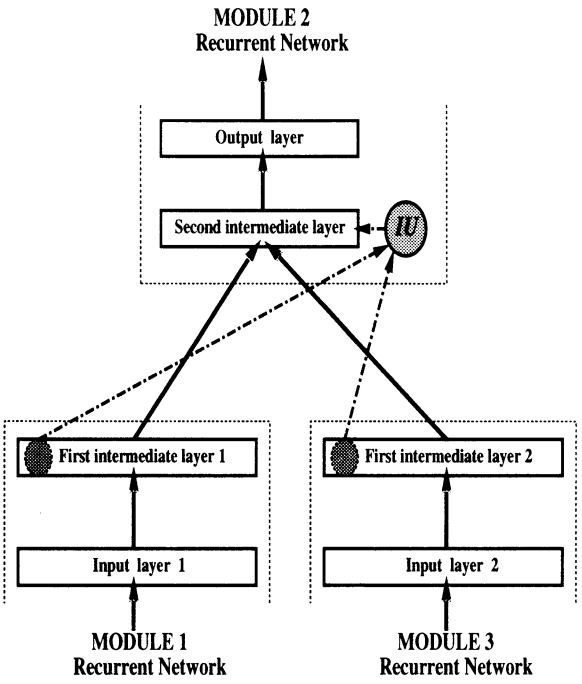


Figure 3: Construction of ξ-connection

patterns of the other modules). The Recurrent network decomposes superposed pattern recalled in feed-forward networks into each pattern and recalls all matched patterns sequentially.

The intermodular connections of MuNet are constituted by connections from the output-layers of some modules to the input-layer of an module and the modules connect each other. The three-layer Back-Propagation network, which is constituted by the two-layer feed-forward network in each module and the output-layers connecting to it, learns interdependency among modules.

Thus this three-layer network of intermodular connection learns interdependency among modules, and it is difficult that the part relating to additional module processes effectively the change of the interdependency when some modules are added. So it is necessary to improve intermodular connection.

Then we improved this intermodular connections structure.

3. Intermodular Connection Network ξ-connection

3.1. General View of ξ-connection

In the intermodular connection described in Sec. 2, the module which receives outputs of other modules is called *input-side module* and the module which send its output to the input-side module is called *output-side module.*

An effect caused by module addition is the change of the learning patterns. Especially intermodular connections whose input-side modules include additional modules, have to be considered.

Table 1: An example of associative patterns

MODULE 1	MODULE 2	MODULE 3
3	△	○
4	□	
	▭	×
	◇	

In such intermodular connections, it is possible that some learning patterns can be correctly recalled by using only connections constructed before module addition. So it is not necessary to relearn for the network.

Therefore, it can be considered that by using the original connections the waste of learning is decreased and the learning speed increases. ξ-connection has merit that the original network constructed before addition of module is not changed and only the additional network constructed after module addition learns learning patterns by using the original network, and can process an effect caused by module addition.

3.2. Change of Learning Pattern caused by Module Addition

An example of interdependency is described in Table 1.

When MODULE 1's output is the pattern "4" and MODULE 3's output is "×", MODULE 2 receives the superposed pattern constituted by the patterns "square" and "diamond" as input. Then recurrent network decomposes the superposed pattern to each patterns. While in case that MODULE 1's output is "4" and MODULE 3's output is no pattern, MODULE 2 receives the superposed pattern constituted by the patterns "square", "rectangle" and "diamond".

The changes of learning patterns are classified into two types. One is the type to decrease the number of the patterns superposed and another is the type to increase. And some learning patterns do not need to change the number of patterns superposed.

3.3. Structure of ξ-connection

ξ-connection is the four-layer network which is constituted by input-layer, first and second intermediate-layer, and output-layer.

Input-layer transform the output of the recurrent network of each module, whose value is binary value { 0, 1 }, into { -1, 1 }. The output function of input-layer is described in Eq. (1).

$$f_i(x) = \frac{2}{1 + e^{-a(x-0.5)}} - 1 \qquad (1)$$

where x is input value and a is the coefficient of sigmoid (a = 10000.0).

First intermediate-layer is the output layer of output-

side modules. In this layer, each unit learns the pattern to express "no pattern" or one of patterns to learn for the module. Then this layer recognizes input pattern and transforms the pattern into symbol by using function Eq. (2).

$$f_{m1}(\mathbf{x}) = \frac{1}{1+e^{-a(\frac{(\mathbf{W}^{m1})^T\mathbf{X}}{c\sqrt{\mathbf{X}^T N\mathbf{X}}} - \sqrt{\frac{p-2}{2(p-1)}})}} \tag{2}$$

where $\mathbf{x}$ is the vector of a input pattern, a is the coefficient of sigmoid, N is the transform matrix and p is the number of patterns. $\mathbf{w}^{m1}$ and c are described in Eq. (3) and Eq. (4).

$$\mathbf{w}^{m1} = (\mathbf{x}')^T N \tag{3}$$

$$c = \sqrt{\mathbf{x}' N \mathbf{x}'} \tag{4}$$

where $\mathbf{x}'$ is the vector of the pattern which a unit correspond to.

Second intermediate-layer is the input-layer of output-side module and its output the value of function Eq. (5).

$$f_{m2}(\mathbf{x}, x_r) = \frac{1}{1+e^{-a((\mathbf{W}^{m2})^T\mathbf{X}+x_r-0.5)}} \tag{5}$$

where $\mathbf{x}$ is the vector of the input from first intermediate-layer, a is the coefficient of sigmoid and x_r is the input from a inhibitory unit. $\mathbf{w}^{m2}$ is connection weight vector between first and second intermediate layer.
In this layer, the patterns which is the output of the recurrent network of the output-side module, are recalled as symbols.

Output-layer transform the symbols that is second intermediate layer's outputs, into a pattern by using function Eq. (6).

$$f_o(\mathbf{x}) = \frac{1}{1+e^{-a(\mathbf{W}^{out})^T\mathbf{X}-0.5}} \tag{6}$$

where $\mathbf{x}$ is the vector of the input from first intermediate-layer and a is the coefficient of sigmoid. $\mathbf{w}^{out}$ is connection weight vector between second intermediate layer and output-layer.

Inhibitory unit (IU) inhibit all unit of second intermediate-layer when the outputs if all input-side modules are "no pattern". The function of IU is described in Eq. (7).

$$f_{iu}(\mathbf{x}) = \prod_i x_i \tag{7}$$

where $\mathbf{x} = (x_i)$ is the output of first intermediate-layer's units that correspond to "no pattern".

3.4. Learning Algorithm of ξ-connection

ξ-connection learns by using the algorithm described below.

First is the learning between input-laye.' and first intermediate-layer. $\mathbf{w}^{m1}$ and matrix N are calculate by using Eq. (3), Eq. (8) and Eq. (9).

$$N = (X \mid Z)^{-T} \begin{pmatrix} Y^T Y & \mid & 0 \\ 0 & \mid & 0 \end{pmatrix} (X \mid Z)^{-1} \tag{8}$$

$$< \mathbf{y}_i, \mathbf{y}_j > = \begin{cases} c^2 & i = j \\ -\frac{c^2}{p-1} & i \neq j \end{cases} \tag{9}$$

where X is the matrix whose columns are the vector of input patterns to learn, Z is the matrix whose columns are the perpendicular vector to every column vectors of X. c is the norm of the column vectors of X and p is the number of learning patterns.

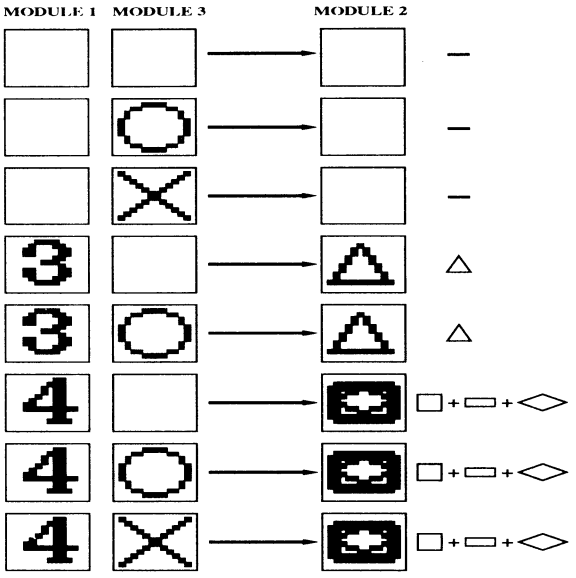


Figure 4: Recollection before second learning

Second is the learning between second intermediate-layer and output-layer. $\mathbf{w}_{ij}^{out}$ is modulated 1 when both of i th second intermediate-layer unit and j th output-layer unit are active in "two direction learning".

Third is the learning between first and second intermediate-layer. $\mathbf{w}^{m2}$ is modulated on the basis of associative interdependency as symbols.

Finally all weight of inhibitory units is determined 1.

4. Simulation

4.1. Simulation about learning pattern change

Figure 4 and Figure 5 show the results of this simulation to examine the ability to process the effect caused by module addition. In this simulation, we used patterns composed of 14 × 14 bits as input patterns and teacher patterns (similarly in other simulations), and the connection network is described in Figure 3 and the interdependency is described in Table 1.

For a start, the network was constituted by MODULE 1 (input-side) and MODULE 2 (output-side), and learned original learning patterns . Then the number of the first intermediate-layer units of MODULE 1 was 3 and that of the second intermediate-layer units of MODULE 2 was 4. Next MODULE 3 was added to the network, and learned additional learning patterns. Then the original network was not changed and the number of the first intermediate-layer units of MODULE 3 was 3.

Figure 4 shows the result of the recollection before MODULE 3 addition and Figure 5 shows that after the addition.

On account of the results, we could think that ξ-connection is suitable for module addition.

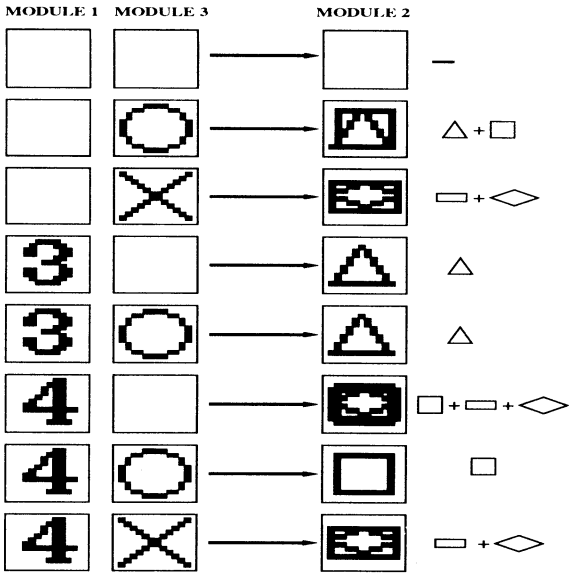


Figure 5: Recollection after second learning

4.2. Comparison with Other Networks about the Degree of Noise Tolerance

Figure 6 shows the result of the comparison the degree of noise tolerance of ξ-connection with that of the intermodular connection of MuNet (relearning) and ν-connection (additional learning and relearning)[3]. ν-connection is a intermodular connection network that has suitable structure for module addition.

In this simulation, the number of MuNet's hidden units

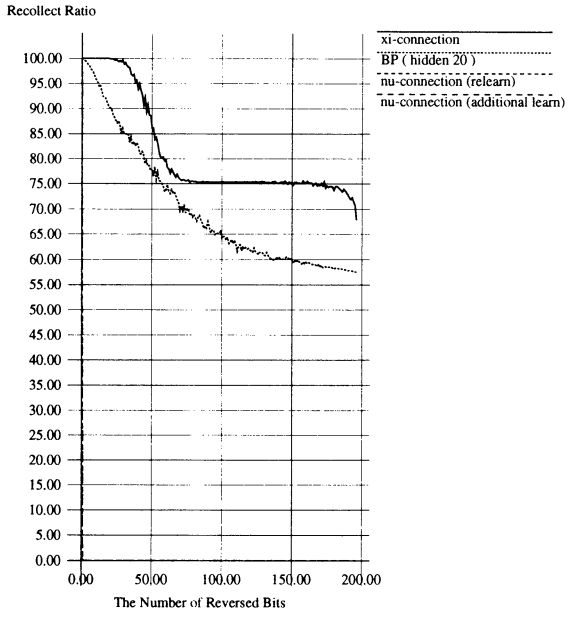


Figure 6: Comparison about the degree of noise tolerance

Table 2: Comparison of learning speed

		Epoch	Learning time [s]
ξ-connection	Additional learning	1.0	7.38
	Relearning	1.0	49.28
The intermodular connection of MuNet	Relearning	1188.2	160.72

was 20 and the pattern described Table 1 was used as the associative interdependency. The network structure of ξ-connection was similar to the simulation Sec. 4.1.

Figure 6 shows that ξ-connection has higher degree of noise tolerance than other three networks (similarly in cases of other ten associative patterns). On account of the results, we could think that ξ-connection has a high degree of noise tolerance.

4.3. Comparison of Learning Speed with MuNet

Table 2 shows the result of the comparison the learning speed of ξ-connection (additional learning and relearning) with that of MuNet's intermodular connection (relearning) on SparcStation 5. In this simulation, the number of MuNet's hidden units was 20 and ten associative patterns that have the interdependency as Table 1 was examined. It is difficult to do additional learning for MuNet's intermodular connection of MuNet.

This result shows that ξ-connection has higher learning speed than MuNet's intermodular connection. According to this result, it was shown that ξ-connection was adequate for module addition and additional learning using originary network is a efficient learning way.

5. Conclusion

We proposed a new connection model of multimodular associative neural networks ξ-connection, which needs to learn in the only connection weights between existing and additional modules and has high degree of noise tolerance.

By the simulation we showed that ξ-connection is suitable for module addition and has high degree of noise tolerance.

References

1. M. Jordan and R. Jacobs(1994), Hierarchical mixtures of experts and the em algorithm, Neural Computation, vol.6, pp.181-214

2. R. Jacobs and M. Jordan(1991), Adaptive Mixtures of Local Experts, Neural Computation, vol.3, no.1, pp.79-87

3. M. Mochizuki, T. Ohsumi, M. Kajiura and Y. Anzai(1994), Multimodule associative neural networks suitable for module addition, Technical report of the Institute of Electronics, Information and Communication Engineers, vol. 94, no. 15, pp. 63 - 70

4. T. Ohsumi, M. Kajiura and Y. Anzai(1993), Multimodule Neural Networks for Associative Memory, Systems and Computers in Japan, vol. 24, no. 13, pp. 98 - 108

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Multi-Winners Self-Organizing Multidirectional Associative Memory

Jiongtao Huang and Masafumi Hagiwara
Department of Information and Computer Science, Keio University, Yokohama, Japan 223-8522
huang@soft.ics.keio.ac.jp

Abstract

We propose a new Multi-directional Associative Memory (MAM) named Multi-Winners Self-organizing MAM (MWS-MAM) based on the distributed representation paradigm. The proposed MWS-MAM is constructed by plural Multi-Winners Self-Organizing Neural Networks (MWSONNs). The distributed representation layers of MWSONNs are fully connected. Each weight matrix existing between any two distributed representation layers is trained by Hebbian Learning, and the weights of MWSONN are trained by Error Correction Learning. The MWS-MAM can represent the input scenes of events as the distributed representations and can store the relations of scenes each other. The MWS-MAM can store and recall plural episodes whose scenes can be represented as analog patterns and it can store episodes containing the multi-looped structure.

1 Introduction

The neural network associative memories[2-10] are one of the most important paradigms for realizing the neuro-based computer. In order to realize a brain-liked computer, the following characteristics of an associative memory can be considered to be very important.

(1) It can realize auto-association, hetero-association, and episodic-association based on same architecture.
(2) It can realize many-to-many associations.
(3) It can create the representation for analog or two-valued stored information automatically.

The Associatron[2] and the BAM[3] are the basic models which can realize the two-valued patterns auto-association and hetero-association. The Pseudo-Relaxation Learning Algorithm for BAM (PRLAB)[4] and the Quick Learning for BAM (QLBAM)[5] have been proposed for increasing the storage capacity of BAM. The PRLAB and the QLBAM can realize the two-valued association only. In order to use the QLBAM to realize episodic-association, the Episodic Associative Memories (EAMs)[6] have been proposed. The EAMs use Pseudo-Noise (PN) sequences[6] and QLBAM which enables high memory capacity. However, it can deal with only the two-valued patterns. In addition, the robustness for internal damage of EAMs is very weak since the PN sequences are employed. In addition, the above mentioned models can not deal with many-to-many associations. The MAM[7] is a successful model which can realize many-to-many association. Since the Habbian Learning is employed, it has the above mentioned problems.

In order to realize the above mentioned abilities (1) ~(3). We propose a new associative memory named Multi-Winners Self-organizing Multi-directional Associative Memory (MWS-MAM) based on the distributed representation paradigm[1]. The MWS-MAM is constructed by 3 or more Multi-Winners Self-Organizing Neural Networks (MWSONNs)[8]. The distributed representation layers of the MWSONNs are fully connected by MAM weight matrices. Each MWSONN has an input-output layer and a distributed representation layer, and these two layers are fully connected by upward and downward weights, respectively. Since the MWSONN can realize the auto-association[8], hetero-association[9], and episodic-association[10], we use the MWSONNs to create the distributed representations for the stored information and to store the primitive associative structures at first. Then, we use the MAM to store the relations of each primitive associative structure. Thus the proposed MWS-MAM can realize many-to-many association. In addition, since the creation of the distributed representation is based on the multi-winners competition, the proposed MWS-MAM can create the representation for the analog or two-valued stored information automatically[8].

This paper is organized as follows. In **2**, the MAM is explained briefly. In **3**, the proposed MWS-MAM is explained in detail. The computer simulation results are shown in **4**.

2 Multidirectional associative memory

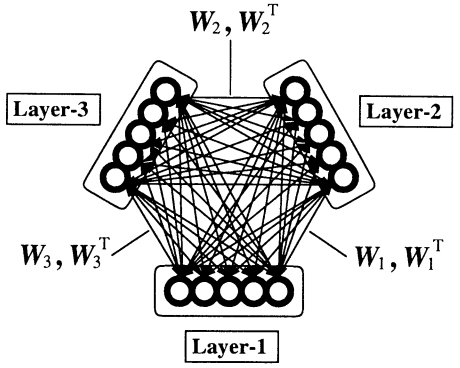


Figure 1: Structure of the MAM.

The conventional Multidirectional Associative Memory (MAM)[7] can be considered a generalized Bidirectional Associative Memory (BAM)[3] because the MAM can store multiple association such as $(\boldsymbol{A}^{(i)},\boldsymbol{B}^{(i)},\boldsymbol{C}^{(i)},\cdots)$. Fig.1 shows the simplest three-layered MAM.

We use the MAM to store the following training set to explain the storage phase of the MAM.

$$\{(\boldsymbol{A}^{(i)},\boldsymbol{B}^{(i)},\boldsymbol{C}^{(i)})\} \quad \text{for} \quad i=1,2,\cdots,p \tag{1}$$

where each training vector is represented by a two-valued bipolar mode, and p is the number of the training data to be stored in the MAM. Let $\boldsymbol{W}_1$ be the weight matrix between neurons in the Layer-1 and the Layer-2. The $\boldsymbol{W}_1$ is trained by Habbian Learning as follows,

$$\boldsymbol{W}_1 = \sum_{i=1}^{p} \boldsymbol{A}^{(i)T}\boldsymbol{B}^{(i)}. \tag{2}$$

The weight from the Layer-2 to Layer-1 are represented by the transpose matrix $\boldsymbol{W}_1^T$,

$$\boldsymbol{W}_1^T = \sum_{i=1}^{p} \boldsymbol{B}^{(i)T}\boldsymbol{A}^{(i)}. \tag{3}$$

Thus, the MAM can store the p training pairs: $(\boldsymbol{A}^{(1)},\boldsymbol{B}^{(1)}),\cdots,(\boldsymbol{A}^{(p)},\boldsymbol{B}^{(p)})$.

The other weight matrices are learned by similar algorithm and represented by $\boldsymbol{W}_2$ (Layer-2 $\Rightarrow$ Layer-3), $\boldsymbol{W}_2^T$ (Layer-3 $\Rightarrow$ Layer-2), $\boldsymbol{W}_3$ (Layer-3 $\Rightarrow$ Layer-1), $\boldsymbol{W}_3^T$ (Layer-1 $\Rightarrow$ Layer-3).

For the recall phase of the MAM, each neuron receives its input from the neurons in the other layers. Let a_i, b_i, and c_i be the output of the i-th neuron in the Layer-1, Layer-2, and Layer-3, respectively, then the state of each neuron is determined by the following equations,

$$\begin{aligned} a_i &= \psi(\sum_j (W_{(1)ij}b_j + W_{(3)ij}c_j)) \\ b_i &= \psi(\sum_j (W_{(2)ij}c_j + W_{(1)ij}a_j)) \\ c_i &= \psi(\sum_j (W_{(3)ij}a_j + W_{(2)ij}b_j)) \end{aligned} \tag{4}$$

where the $W_{(1)ij}$, $W_{(2)ij}$, and $W_{(3)ij}$ denote the component of the matrices $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, and $\boldsymbol{W}_3$, respectively.

The MAM has the following features owing to its structure[7],

- It can recall many other data by receiving one key input (many-to-many association).
- When an interference from the other layer or noise exists, it can be suppressed by recalling the other data (robustness against noise).
- Recall becomes correct by memorization with the other data (robust memory by a context).

3 Multi-winners self-organizing multidirectional associative memory

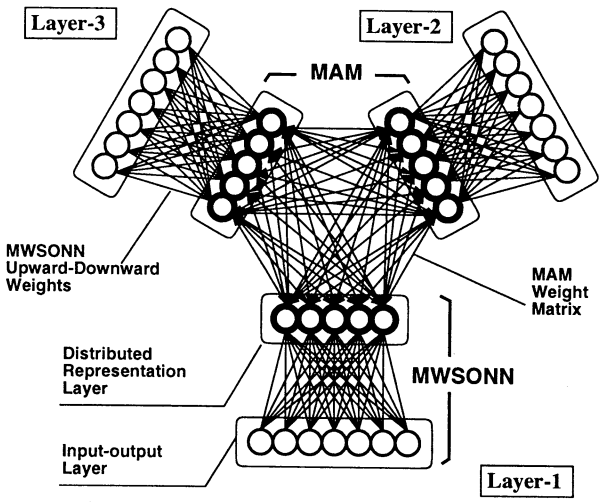


Figure 2: Structure of the MWS-MAM.

Structure of the proposed Multi-Winners Self-organizing Multidirectional Associative Memory (MWS-MAM) is shown in Fig.2. The proposed MWS-MAM is constructed by three Multi-Winners Self-Organizing Neural Networks (MWSONNs)[8]. And the distributed representation layers of MWSONNs are fully connected. Each MWSONN has an input-output layer and a distributed representation layer. The detail structure of the MWSONN is shown in Fig.3.

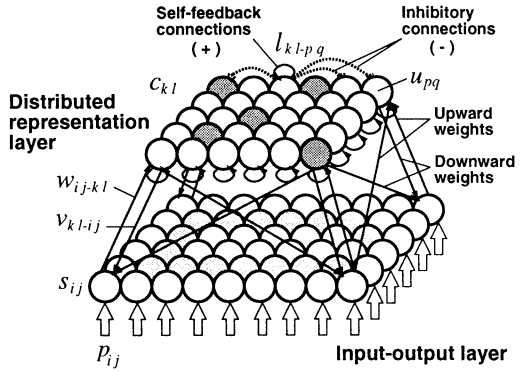


Figure 3: Detail of the MWSONN.

The neurons in the distributed representation layer are connected by the self-excitatory, neighbor-inhibitory connections l_{kl-pq}, the neurons in the input-output layer and the neurons in the distributed representation layer are connected by upward weights w_{ij-kl} and downward weights v_{kl-ij}. The neurons in the input-output layer receive the input p_{ij} and output s_{ij} through the ramp function,

$$S_f(x) = \begin{cases} x & ; \quad x > 0 \\ 0 & ; \quad x \leq 0. \end{cases} \tag{5}$$

We employ the ramp function in the transfer function of the input-output neurons in order to process analog information.

The neurons in the distributed representation layer receive the output from the input-output layer and the output of the other neurons in the distributed representation layer through the following sigmoid function,

$$C_f(x) = \frac{1}{1 + \exp(-x/T)}. \tag{6}$$

The strength of the connection l_{kl-pq} can be calculated by,

$$\begin{cases} l_{kl-pq} = \Psi(d_{kl,pq}) \\ \Psi(x) = (A - Bx^2)\exp(-x^2/\sigma^2) \\ d_{kl,pq} = \sqrt{(k-p)^2 + (l-q)^2} \end{cases} \tag{7}$$

where, A, B, and σ are the parameters which decide the shape of the neighborhood function $\Psi(\cdot)$.

3.1 Storage phase of the MWS-MAM

The storage phase of the MES-MAM can be separated by two steps: (1) The MWSONN storage; (2) The MAM storage. For the MWSONN storage, each MWSONN receives the analog input from the input-output layer and creates the distributed representation of current input in the distributed representation layer by the following multi-winners competitive dynamics[8],

$$\begin{cases} \tau \dfrac{d}{dt} u_{kl} = -u_{kl} + \displaystyle\sum_{p=1}^{P}\sum_{q=1}^{Q} l_{pq-kl} \cdot c_{pq} \\ c_{kl} = C_f(u_{kl} - \theta_{kl}) \\ c_{kl}^{(0)} = C_f\left(\displaystyle\sum_{i=1}^{I}\sum_{j=1}^{J} w_{ij-kl} \cdot s_{ij} - \theta_{kl}\right) \\ s_{ij} = S_f(p_{ij}) \end{cases} \tag{8}$$

where u_{kl} is the potential, c_{kl} is the output, and θ_{kl} is the threshold of the klth neuron in the distributed representation layer. After the representation is created, the MWSONN stores the relations between input and its presentation into the upward and downward weights by the following error correction learning.

$$\begin{cases} \tau_w \dfrac{dw_{ij-kl}}{dt} = -w_{ij-kl} \\ \qquad\qquad +\alpha a_{ij}(a'_{kl} - a_{kl}^{(comp)})a'_{kl}(1 - a'_{kl}) \\ \tau_v \dfrac{dv_{kl-ij}}{dt} = -v_{kl-ij} + \beta a_{kl}^{(comp)}(b_{ij} - b_{ij}^{(orig)}) \end{cases} \tag{9}$$

where, a_{ij} is the component of input, $a_{kl}^{(comp)}$ is the component of representation of input, $b_{ij}^{(orig)}$ is the component of the other related input, a'_{kl} is the output component of distributed representation layer and b_{ij} is the output component of input-output layer. α and β are the learning rates, τ_w and τ_v are the time delay factors.

Since the MWSONN can store analog patterns and it can realize the auto-association, hetero-association, and episodic-association based on the similar storage algorithms[9][10]. Each MWSONN can store plural primitive association structures.

After the MWSONN storage is completed, the MAM part of the proposed MWS-MAM stores the relations of each primitive association structures which are stored in each MWSONN by storing the relations of the distributed representations. The training algorithm of the MAM is shown in Eq.2 and Eq.3.

3.2 Recall phase of the MWS-MAM

In the recall phase, first the input-output layers receive key input. Then several neurons in the distributed representation layers are fired. And these activations can be propagated to the other distributed representation layers of MWSONNs by MAM. Based on the reverberations which are occurred in each MWSONN, the stored primitive association structures can be recalled. And based on the reverberations which are occurred in the MAM, the MWS-MAM can recall

the other primitive association structures from a key input.

4 Computer simulation

4.1 Simulation conditions

In order to examine the behavior of the proposed MWS-MAM, we use the proposed MWS-MAM to store the analog pattern pairs. The simulation conditions are summarized in Table.1. Fig.4 shows the three stored triple pattern pairs.

Table 1: Conditions for Simulations

Size of the patterns	11×11 pixels
Input-output layers	11×11 neurons
Distributed representation layers	7×8 neurons
Number of winners	about 15 neurons
Component of patterns	[0,1] analog valued

4.2 Simulation results

Under the above mentioned conditions, we use the proposed MWS-MAM to store the following three triple analog pattern pairs,

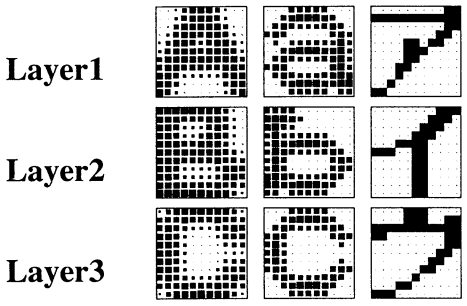


Figure 4: Three stored triple pattern pairs, (A,B,C), (a,b,c), and (ア, イ, ウ).

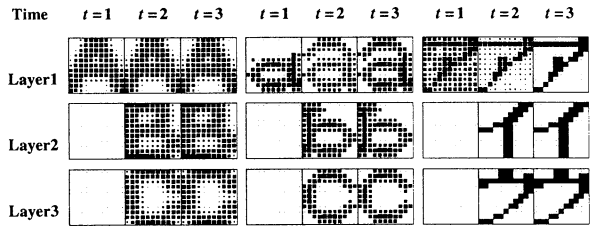


Figure 5: Recall from failed or noisy key input.

Fig.5 shows the recall results of the trained MWS-MAM. We can find that the proposed MWS-MAM model can store the analog valued pattern pairs and it also can recall the related stored pattern pieces from the incomplete key input or the noisy key input.

5 Conclusion

We have proposed a new associative memory named MWS-MAM in this paper. The proposed MWS-MAM has the following features owing to its structure:

- It can realize auto-association, hetero-association, and episodic-association based on same architecture.
- It can realize many-to-many associations.
- It can create the representation for analog or two-valued stored information automatically.

References

[1] D. E. Rumelhart, J. L. McClelland and the PDP Research Group, *Parallel Distributed Processing*, Vol.1 and Vol.2, MIT Press, 1989.

[2] Nakano, *Associatron*, Syokodou, 1988.

[3] B. Kosko, "Bidirectional Associative Memories," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol.18, No.1, pp.49-60, 1988.

[4] H.Oh and S. C. Kothari, "Adaptation of the relaxation method for learning in bidirectional associative memory," *IEEE Trans. on Neural Networks*, Vol.5, No.4, pp.576-583, 1994.

[5] M. Hattori and M. Hagiwara, "Quick Learning for Bidirectional Associative Memory," *IEICE Trans. Inf. & Syst.*, Vol.E77-D, No.4, pp.385-392, 1994.

[6] M. Hattori and M. Hagiwara, "Episodic Associative Memories," *Neurocomputing*, Vol.12, pp.1-18, 1996.

[7] M. Hagiwara, "Multidirectional associative memory," *IJCNN'90* (Washington, D. C.), Vol.1, pp. 3-6, 1990.

[8] J. T. Huang and M. Hagiwara, "A multi-winners self-organizing neural network," *Trans. of the IEICE*, **D-II**, Vol.J81, No.3, pp.547-556, 1998.

[9] J. T. Huang and M. Hagiwara, "Multi-winners self-organizing bidirectional associative memory," *1998 IEEE Inter. Conf. on Systems, Man, and Cybernetics*, (San Diego), pp.3641-3646, 1998.

[10] J. T. Huang and M. Hagiwara, "Multi-winners self-organizing episodic associative memory," *1998 IEEE Inter. Conf. on Systems, Man, and Cybernetics*, (San Diego), pp.3635-3640, 1998.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Bill Money Recognition by Using the LVQ Method

Toshihisa Kosaka and Sigeru Omatu
Glory Ltd. Osaka Prefecture University
Himeji, Hyougo 670-8567 Sakai, Osaka 599-8531,
Japan Japan

Abstract The progress of the computer science enables us to process complex and large scale computation and advanced pattern recognition methods could be adopted to process the pattern classification problems. Among them neuro-pattern recognition which means the pattern recognition based on the neural network approach has been paid an attention since it has classified various patterns like human beings. In this paper, we adopt the learning vector quantization(LVQ) method to classify the various money. The reasons to use the LVQ are that it can process the unsupervised classification and treat many input data with small computational burdens. We will construct the LVQ network to classify the Italian Liras. Compared with a conventional pattern matching technique, which has been adopted as a classification method, the proposed method has shown excellent classification results.

1. Introduction

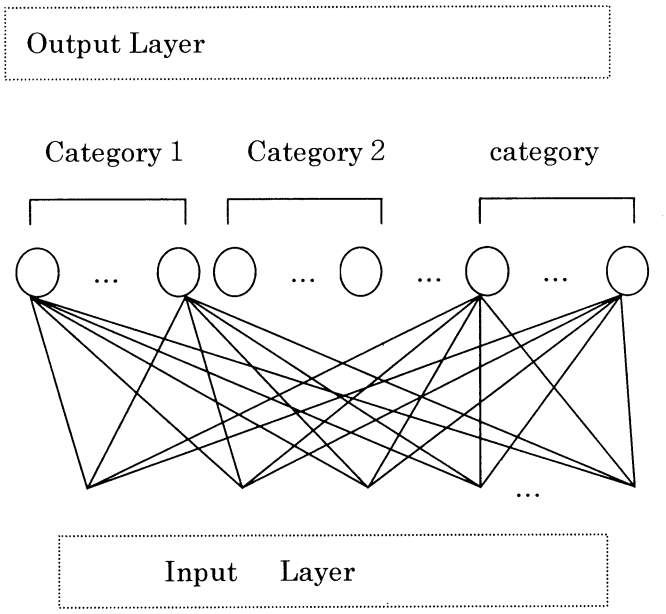
Bill money classification by transaction machines has been important to make progress the office automation[1]. Since sizes of bills are different according to kinds of bills, the measurement data of bills include various variations. Human being can classify the bills correctly even if they are suffered from those variations such as rotation and shift. But usual pattern recognition using a conventional transaction machine cannot give us the correct classification result under such cases since the basic method is a pattern matching principle. Furthermore, the conventional pattern matching method requires many template patterns for many kinds of bills, which takes much time and needs much experience[1].

Recently, neural networks which are based on the biological mechanism of human brain have been focussed since they have intelligent pattern recognition ability[2]. In this paper, we will apply the neural network approach to classify the bill money under various conditions by using transaction machines. The learning vector quatization(LVQ) has been used to classify the bills since it can treat high dimensional input and has simple learning structure[3]. The LVQ network adopted here has 64x15 units in the input layer and many units at the output layer. The bills are Italian Liras of of 8 kinds, 1,000, 2,000, 5,000, 10,000, 50,000(new), 50,000(old), 100,000(new), 100,000(old) Liras with four directions A,B,C, and D where A and B mean the normal direction and the upside down direction and C and D mean the reverse version of A and B. The simulation results show that the proposed method can produce the excellent classification results.

2. Competitive Neural Networks

We will explain the competitive neural networks

that are used to classify the bill money. The structure of a LVQ competitive network is shown in Figureure 1. The input for the LVQ is bill money data where an original image consists of 128x64 pixels and the input data to the network is compressed as 64x15 pixels to decrease the computational load. The output of the network consists of the Italian Liras of 8 kinds, 1,000, 2,000, 5,000, 10,000, 50,000(new),50,000(old),100,000(new), 100,000(old) Liras with four directions A,B,C, and D where A and B mean the normal direction and the upside down direction and C and D mean the reverse version of A and B.



Figureure 1. LVQ networks.

In the input layer the original bill money data are applied and all the units at the input layer are connected to all the neurons at the output layer with connection weight W_{ij}. W_{ij} denotes the connection weight from unit j at the input layer to unit I at the output layer. The connection weights are set by the random number at the beginning. Then the following learning algorithm of the connection weight vector is used in the LVQ method.

Step 1. Find the unit c at the output layer which has the minimum distance from the input data **x** (t)

$$\|\mathbf{x}(t) - \mathbf{W}_c\| = \min_i \|\mathbf{x}(t) - \mathbf{W}_i\|$$

where $\| \ \|$ denotes the Euclidean norm and t denotes iteration time.

Step 2. If the input **x** (t)belongs to Category c, then

$$\mathbf{w}_c(t+1) = \mathbf{w}_c(t) + \alpha(t)(\mathbf{x}(t) - \mathbf{w}_c(t))$$
$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t), \quad i \neq c$$

and if the input **x** (t) belongs to the other Category j ($j \neq c$), then

$$\mathbf{w}_c(t+1) = \mathbf{w}_c(t) - \alpha(t)(\mathbf{x}(t) - \mathbf{w}_c(t))$$
$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t), \quad i \neq c$$

where $\alpha(t)$ is a positive function and denotes learning rate. $\alpha(t)$ of the usual LVQ is given by

$$\alpha(t) = \alpha_0(1 - \frac{t}{T})$$

where α_0 ($0 < \alpha_0 < 1$) is a positive constant and T is a total number of learning iterations.

Step3. Input a new pattern **y**(t) and **y**(t) $\Rightarrow$**x**(t). Goto *Step1*.

The above algorithm can be explained graphically as Figure 2.

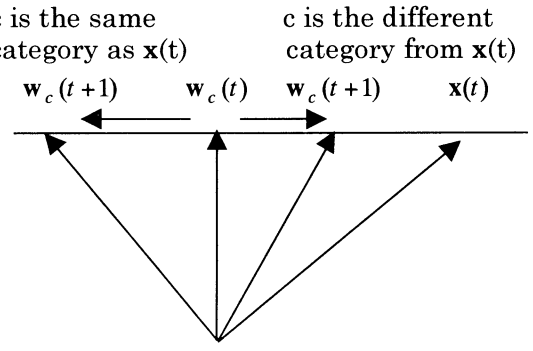


Figure 2. Principle of the LVQ algorithm where the right hand side shows the same category case of **x**(t) and Category c and the left hand side denotes the different category case.

In the above LVQ algorithm, the learning rate $\alpha(t)$ plays important role for convergence. To adjust the parameter, Kohonen has proposed an optimization method without proof as follows:

$$\alpha_c(t) = \frac{\alpha_c(t-1)}{1+s(t-1)\alpha_c(t-1)}$$

where s(t) =1 if $\mathbf{x}(t)$ belongs to the same Category c and s(t)=-1 if $\mathbf{x}(t)$ does not belong to the same Category c. Here, $\alpha_c(t)$ denotes the learning rate $\alpha(t)$ for the pattern from Category c.

In what follows, we will prove the above relation.

From the learning rule of the LVQ, we have

$$\begin{aligned}\mathbf{w}_c(t+1) &= \mathbf{w}_c(t) + s(t)\alpha_c(t)(\mathbf{x}(t) - \mathbf{w}_c(t)) \\ &= (1-s(t))\alpha_c(t)\mathbf{w}_c(t) + s(t)\alpha_c(t)\mathbf{x}(t)\end{aligned}$$

and

$$\begin{aligned}\mathbf{w}_c(t) &= \mathbf{w}_c(t-1) + s(t-1)\alpha_c(t-1)(\mathbf{x}(t-1) - \mathbf{w}_c(t-1)) \\ &= (1-s(t-1))\alpha_c(t-1)\mathbf{w}_c(t-1) + s(t-1)\alpha_c(t-1)\mathbf{x}(t-1)\end{aligned}$$

Substituting the latter equation the former one, we have

$$\begin{aligned}&\mathbf{w}_c(t+1) = (1-s(t)\alpha_c(t))(1-s(t-1)\alpha_c(t-1))\mathbf{w}_c(t-1) \\ &+ s(t)\acute{a}_c(t)\mathbf{x}(t) \; + s(t-1)\acute{a}_c(t-1)(1-s(t)\acute{a}_c(t))\mathbf{x}(t-1).\end{aligned}$$

We assume that the optimal rate adjusts the effect of x(t) and x(t-1) equally within the absolute value, that is,

$$\alpha_c(t) = (1-s(t)\alpha_c(t))\alpha_c(t-1).$$

Then we have

$$\alpha_c(t) = \frac{\alpha_c(t-1)}{1+s(t-1)\alpha_c(t-1)}.$$

From the above equation, we can see that the value of $\alpha_c(t)$ become larger than 1 when s(t-1)= - 1, which may make the learning algorithm unstable. Thus, we must fix the $\alpha_c(t)$ to a boundary value α_0 when it becomes larger than 1.

$$\alpha_c(t+1) = \alpha_0 \quad \text{if} \quad \alpha_c(t+1) > 1.$$

Using the above OLVQ1 algorithm, we will classify the Italian bills in the following section.

3. Preprocessing Algorithm

The images obtained by transaction machine, there are variations such as rotation or shift. Therefore, we must adjust the images such that the variations may be reduced as much as possible by using the preprocessing. The flow char of the preprocessing procedure is illustrated in Figure 3. In this Figureure, the original image with 128x64 pixels are observed at the transaction machine in which rotation and shit are included. After correction of these effects, we select a suitable aria which show the bill image and compressed as the image with 64x15 pixels to the neural networks. Although the neural network of the LVQ type could process any order of the dimension of the input data, the small size is better to achieve the fast convergence result. Thus, we have selected the above size of the image.

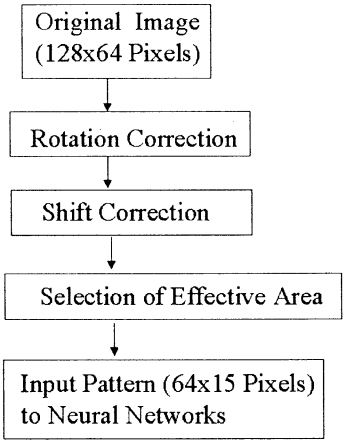


Figure 3. Preprocessing algorithm.

4. Italian Lira Classification Algorithm

The bills used here are Italian liras, which have 8 kinds such as 1,000 Liras, 2,000 Liras, 5,000 Liras, 10,000 Liras, new 50,000 Liras, old 50,000 Liras, new 100,000 Liras, and old 100,000 liras. Those Lira bills are used at the input of the transaction

machine where four directions such as A, B, C, and D appear since normal direction, reverse direction, and their upside down directions occur at the input. Therefore, thirty-two bills images are one set of the classification pattern of the experiment. Total number of data sets is 30 and 10 data sets are used for training of the network and the remainig 20 data sets are used to test the network. The parameter used here are α_0 =0.5, T=150, the number of the neurons at the competitive layer=32 at the first time, and the threthold value for classification d_θ=1,000.

Afterb training the neural network, 20 data sets are tested how well the LVQ network could work. Table 1 denotes the classification results by using a conventianal pattern matching method. Table 2 show the classification results by using the proposed method. The classification results are almost the same for both algorithms. But the unfired patterns in Table 1 ismuch more than those of Table 2. Therefore, the proposed method can offer the suitable classification results for Italian Liras.

6. Conclusions

We have proposed a new classification method of Italian Liras by using the OLVQ1 algorithm. The experimental results show the effectiveness of the proposed algorithm compared with the conventional pattern matching method.

References

[1] S. Fukuda, T. Kosaka, and S. Omatu: Bill Money Classification of Japanese Yen Using Time Series Data, Trans. Of IEE of Japan, Vol.115-C, No.3, pp.354-360, 9995(in Japanese).

[2] J. Dayhoff: Neural Network Architectures: An Introduction, International Thompson Computer Press, New York, 1990.

[3] T. Kohonen: Self-Organizing Maps, Springer, Berlin, 1995.

Table 1. Classification by a convetional pattern matching method where the values enclosed by ( ) denote the numbers of unfired neurons.

		Directions			
		A	B	C	D
Italian Liras	1,000	100 (20)	100 (15)	100 (15)	100 (10)
	2,000	100 (5)	100 (10)	100 (25)	100 (25)
	5,000	100 (15)	100 (20)	100 (5)	100 (0)
	10,000	100 (10)	100 (10)	100 (10)	100 (5)
	50,000 (new)	100 (5)	100 (0)	100 (20)	100 (5)
	50,000 (old)	85 (0)	100 (0)	80 (0)	95 (0)
	100,000 (new)	100 (0)	100 (0)	90 (0)	100 (0)
	100,000 (old)	100 (0)	100 (0)	95 (0)	100 (0)

Table 2. Classification by the proposed method where the values enclosed by ( ) denote the numbers of unfired neurons.

		Directions			
		A	B	C	D
Italian Liras	1,000	100 (5)	100 (0)	100 (5)	100 (0)
	2,000	100 (0)	100 (10)	100 (25)	100 (25)
	5,000	100 (15)	100 (20)	100 (5)	100 (0)
	10,000	100 (10)	100 (0)	100 (0)	100 (5)
	50,000 (new)	100 (5)	100 (0)	100 (0)	100 (0)
	50,000 (old)	85 (0)	100 (5)	80 (0)	95 (0)
	100,000 (new)	100 (0)	100 (0)	90 (0)	100 (0)
	100,000 (old)	100 (0)	100 (5)	95 (0)	100 (0)

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Neuro-Approach for Hard Disk Driver Position Control

Toru Fujinaka, Michifumi Yoshioka, and Sigeru Omatu
Osaka Prefecture University
Sakai, Osaka 599-8531, Japan

Abstract Computational intelligence approaches like fuzzy theory, neural network, genetic algorithm, artificial life, etc. have been well developed and applied to many real control problems in an efficient way. In real control processes, the PID controller has been used as the major control methods and the operators select suitable PID gains from time to time based on their experience and knowledge about the plant dynamics. However, it is required to find suitable PID gains automatically. The main reason is that its structure is simple and PID controllers are robust with noise and parameter variations. To use the PID controller, we must tune the PID gains, which have been determined by trial and error, based on experience and knowledge of experts. In this paper, we propose a tuning method of the PID gains by using neural networks to control the position of hard disk driver.

1. Introduction

The hard disk driver must read the data stored in the track of the disk through magnetic head which has been fixed on the arm. Changing the current of the voice coil motor can control the arm [1]-[3]. The driver position follows the center of the track even if the track will move according to the deviation of the disk. Generally, the track of the disk will rotate at 3,600 rpm or 7,200 rpm and the track density is high. Hence, performance of the hard disk becomes sensitive to deviation of the disk. Using the proposed self-tuning PID controller, we will show the comparable following results to H^{∞} control results by the simple computation without a priori knowledge of the hard disk driver.

2. Hard Disk Driver System

The hard disk drive is shown in Figure 1 where the voice coil motor is the actuator of the head arm.

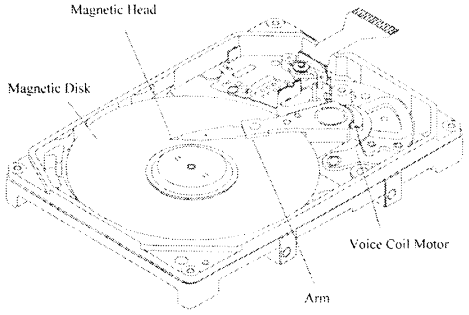


Figure 1. The hard disk drive.

The input output relation can be written as

$$\begin{bmatrix} x_1(n+1) \\ x_2(n+1) \end{bmatrix} = P \begin{bmatrix} x_1(n) \\ x_2(n) \end{bmatrix} + Gu(n)$$

$$y(n+1) = C \begin{bmatrix} x_1(n+1) \\ x_2(n+1) \end{bmatrix}$$

where u(n) denotes input current to the voice coil motor, $x_1(n)$ and $x_2(n)$ are head position and head velocity, respectively, and P, G, and C are constant coefficient matrices given by

$$P = \begin{bmatrix} 0.9987 & 0.0002 \\ -15.6908 & 0.9088 \end{bmatrix}, \quad G = \begin{bmatrix} 0.7096 \\ 8381.7 \end{bmatrix}, \quad C = [1, 0].$$

The above equation has been obtained by sampling the continuous-time system with the sampling frequency of 6kHz. This has been

derived from the fact that the hard disk rotates with 3,600 rpm(=60Hz) and each track has 100 sectors. The aim of the control is to move the head on the specified track. The tracks are located on the circle but the fine structure due to high density of storage and downsizing of the disk will cause the circle not to be precise and rotate on the elliptic circle by the reformation of the disk device. Thus, the head position must follow the cyclic motion with 3,600 rpm. Furthermore, the disk has specific resonance frequency about 2-3 kHz due to the mechanical vibration. Therefore, the control objective has the following three specifications:

1) Following the cyclic motion(servo problem).
2) Stable tracking(robust problem).
3) Reduce the internal vibration.

The conventional control algorithm, for example, feedback control algorithm, adaptive control problem, linear quadratic control problem, etc., could not be applied to this problem since each control algorithm has not satisfied one of these specifications. The only approach in the modern control theory is H□ control method but this approach is difficult to apply to this problem since the controller is required to include the integral action which means the zero is located on the imaginary axis.

Therefore, we apply the control algorithm based on the Proportional+Derivative+Integral (PID) controller that has been adopted in the various application problems. The PID controller has simple structure and the operators can change the PID parameters according the various environments skillfully since they can understand the dynamic effect of PID parameter changes.

3. Neuro-PID Control Systems

The neuro-PID controller that is a PID controller based on the neural network is illustrated in Figure 2.

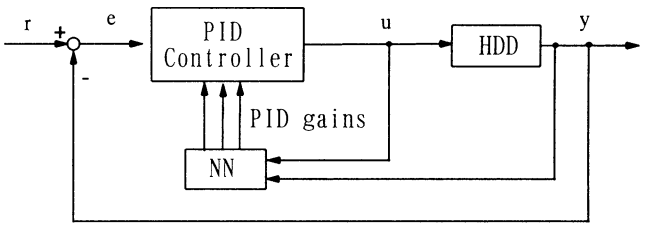


Figure 2. Neuro-PID control systems.

In Figure 2, the neural network can adjust the PID gains just like the human operator based on the input u, output y, and the desired signal r such that the resulting output error may become decrease as much as possible. As we explained before, we must consider the three specifications to design the PID controller. The first one is described by

$$\min_{K_P,K_I,K_D} J_1 = \frac{1}{2}e(n+1)^2 .$$

The second one is to design the controller such that the stable motion can be realized, which can be given by

$$\min_{K_P,K_I,K_D} J_2 = \frac{1}{2}\frac{x_2(n+1)^2}{1+e(n+1)^2} .$$

The third one is to reduce the internal mechanical vibration around 2-3 kHz that can be realized by

$$\min_{K_P,K_I,K_D} J_3 = \frac{|G_{open}(z_0)|}{\gamma}$$

where γ denotes a positive normalizing constant and $G_{open}(z_0)$ denotes the open loop transfer function at $z = z_0$. The open loop transfer function $G_{open}(z_0)$ can be given by

$$G_{open} = (1 + \frac{K_I}{z-1}) G_p(z) G_c(z)$$

where $G_p(z)$ denotes the plant transfer function and $G_c(z)$ is the controller transfer function given by

$$G_c(z) = \frac{K_P(1-z^{-1}) + K_I + K_D(1-2z^{-1}+z^{-2})}{1-z^{-1}}.$$

The neural network used here is three-layered type where the input is the past plant output and input data as well as the desired signal and the output consists of PID gains. The structure of the layered neural network is illustrated in Figure 3.

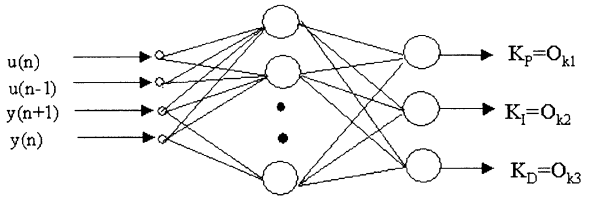


Figure 3. Neuro-tuning scheme.

Using the error back-propagation, we can derive the recursive algorithm to adjust the PID gains from Figure 3. The detail derivation for this algorithm has been reported in [4] except for the different cost function J such that

$$J = J_1 + \gamma_1 J_2 + \gamma_2 J_3.$$

The basic forms of the adjustment rules are

$$\Delta W_{kj}(n+1) = -\eta \frac{\partial J}{\partial W_{kj}} + \alpha \Delta W_{kj}(n), \quad \eta > 0$$

$$\Delta W_{ji}(n+1) = -\eta \frac{\partial J}{\partial W_{ji}} + \alpha \Delta W_{ij}(n).$$

Using the above relation, we can adjust the PID gains and the desired output of the head position can be achieved.

4. Simulation Results

We will consider the following situation:

$z_0 = 2\pi \times 2{,}000$, $\eta = 6 \times 10^{-7}$, $\alpha = 1 \times 10^{-7}$, $\gamma = 5$, $r(n) = \sin(2\pi \times 60n)$.

The remaining parameters are determined based on the simulation results. To find the weighting parameters γ_1 and γ_2, we have changed these parameters as shown in Figure 4.

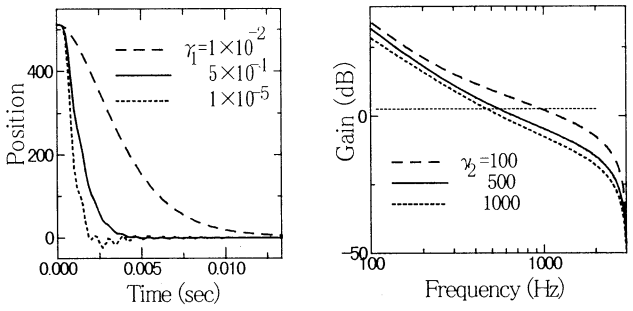


Figure 4. Simulation results for various values of γ_1 and γ_2.

From these results we have determined these parameters as

$$\gamma_1 = 5 \times 10^{-4}, \quad \gamma_2 = 1 \times 10^{4}.$$

In this case, we can get the simulation results as shown in Figure 5 where transient trajectories are illustrated.

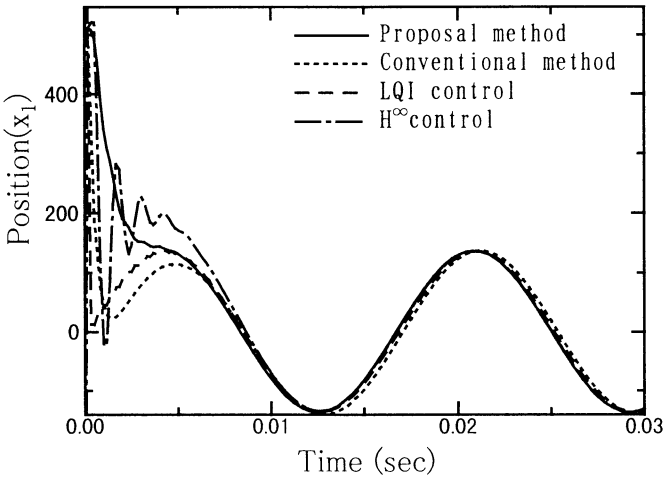


Figure 5. Control results by several control methods.

The H control has been given as in the control scheme shown in Figure 6.

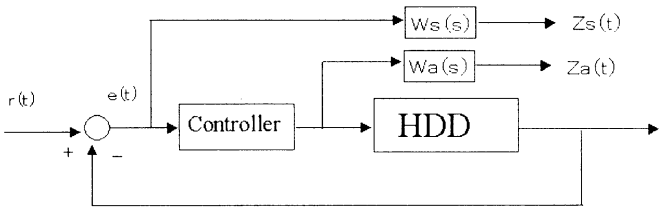


Figure 6. H control scheme used here.

In Figure 6, the transfer functions of $W_s(s)$ and $W_a(s)$ are given by

$$W_s(s) = \frac{2{,}500}{s + 200\pi}, \quad W_a(s) = \frac{5s}{s + 2{,}000\pi}.$$

Figure 7 shows the steady state errors for various methods and Figure 8 shows the gains for them. Here, the conventional method means the adopted method as the hard disk controller and LQI is the integrated linear quadratic control algorithm, which is effective when the desired signal is constant. In Figure 7, the steady state errors of the proposed method is smaller than that by H controller and from Figure 8 the proposed method is more effiective than H controller case since the present method has produced the higher gains in the low frequency and lower gains in the effective frequency band for internal mechanical vibration. From these results we can see that the proposed method has the better control results compared with the other methods.

5. Conclusions

In this paper, we have proposed a new technique to tune the PID control gains of the position control of the hard disk head. Compared with the conventional techniques, the proposed method has shown better control performance with simple structure of the controllers.

References

[1] T. Mita: H Control, Syokodo, Tokyo, 1994(in Japanese).

[2] M. Tomizuka, T. Chin, and C. Kia: Analysis and Synthesis of Discrete-Time Repetitive Controllers, ASME J. Dyn. Syst. Meas. Control, Vol.111, pp.353-358, 1989.

[3] M. Kobayashi, T. Yamaguchi,et al.: Adaptive Control of a Hybrid Servo System for Magnetic Disk Drives, JSME Int. J. , Series C, Vol.39, No.4, pp.772-780, 1996.

[4] S. Omatu, K. Marzuki, and Y. Rubiya: Neuro-Control and Applications, Springer, London, 1995.

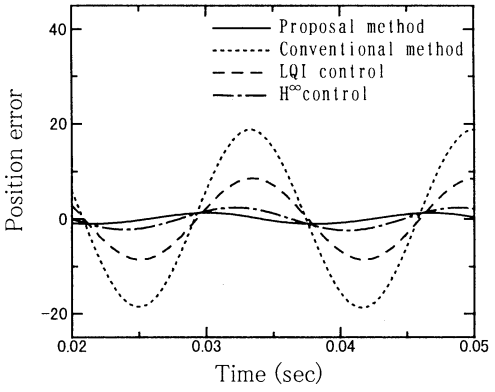


Figure 7. Steady state errors.

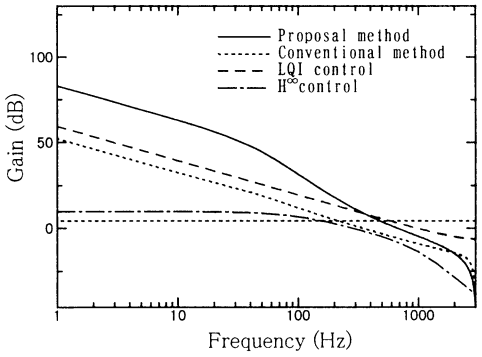


Figure 8. Bode diagram for various methods.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Saccadic Model with Distributed Feedback Mechanism : Simulations of Normal and Interrupted Saccades

Kuniharu Arai[1], Sanjoy Das[2], Edward L. Keller[3], and Eitaro Aiyoshi[4]

[1]Mitsubishi Chemical Corporation
[2]ITT Systems & Sciences Corporation
[3]Smith-Kettlewell Eye Research Institute and University of California, Berkeley
[4]Faculty of Science and Technology, Keio University

abstract

Saccades are rapid and precise eye movements. Recent neurophysiological experiments have showed that saccades are controlled by a closed-loop feedback mechanism. We proposed a 2-D distributed model that placed the superior colliculus inside the feedback loop. Position and velocity signals from brainstem were used for computing the motor error between desired and actual eye position. Parameters in the model were optimized using a GA and recurrent neural net training algorithm. After the optimization, the model was able to produce temporally perturbed saccades accurately as well as normal ones. The results supports the hypothesis that saccades are controlled by a feedback mechanism.

Keywords : Distributed feedback model, saccadic eye movements, interrupted saccades

1. Introduction

Saccade eye movements are among the most rapid yet precise of all movements produced by higher mammals. Initially, it was thought that saccades were controlled by an open-loop feedforward system. However, recent neurophysiological experiments showed that a saccade neural control system used a closed-loop feedback control mechanism in order to generate accurate saccade eye movements. For example, interrupted saccades, saccades that are perturbed in mid-flight by electric stimulations in the omnipause neuron region in monkey, are known to produce final eye displacements with accuracies that are similar to normal saccades[1,2]. The result proved that saccades are actually controlled by a closed-loop feedback system, but left unanswered the question of what type of biological signal was used and what part of the brain participated in computing the motor error between desired (target) and actual eye positions.

Waitzman *et al.* showed that the discharge of SRBN's (saccade related burst neurons) in the superior colliculus (SC) was linearly correlated to the motor error[3], and some saccadic models that placed the SC inside a feedback mechanism were proposed[4,5]. We developed a two-dimensional model of the saccadic system based on neurophysiological experiments, that represented both the SC as a topographically arranged, single-layered array of laterally connected units and brainstem elements including burst generator, neural integrator, and omnipause units[6,7]. A distributed feedback pathway to the SC from the downstream elements, containing both position and velocity components was incorporated in the proposed model.

In the previous paper, we optimized the one-dimensional model (pure horizontal saccadic model) using a hybrid optimization method combining evolutionary computation (genetic algorithm) with a supervised recurrent neural network training algorithm so that the model was able to produce accurate saccades. In this paper, we applied the optimization method to the two-dimensional oblique saccadic model and tested the generation of interrupted saccades. The results supports the hypothesis that saccades are controlled by error feedback of signals that code efference copies of eye motion.

2. Distributed saccadic model

Our model is based on some neurophysiological findings and hypothesis. 1) The SC is placed inside a feedback mechanism and computes the motor error between desired and actual eye movement. 2) Efferent copies of both position and velocity signals from the brainstem are used for computing the motor error. 3) There are lateral inhibitory and excitatory interconnections between units in the SC. Figure 1 shows the proposed model made up of the SC and brainstem models. A unit in the SC receives the visual (target) input (**v**) and the position ($\mathbf{f}_p$) and velocity ($\mathbf{f}_v$) feedback signals, and sends motor commands to the horizontal and vertical burst generators in the brainstem model. The dynamics of the SC units is described by the following equations,

$$\tau_1 \frac{d\mathbf{x}}{dt} = -\mathbf{x} + \mathbf{W}\mathbf{y} + \mathbf{v} - \mathbf{f}_p - \mathbf{f}_v \quad (1)$$

$$\mathbf{y} = f(\mathbf{x}) \quad (2)$$

where **x** and **y** are the activation and output state variables of the SC units and **W** is the lateral inhibitory and excitatory interconnection between units in the SC. The quantity τ_1 is a small time constant and $f(x)$ is a sigmoidal typed nonlinear function. The horizontal and vertical motor commands (m_h and m_v) to the burst generators are calculated by,

$$m_h = \mathbf{w}_h^T \mathbf{y} \quad (3)$$

$$m_v = \mathbf{w}_v^T \mathbf{y} \quad (4)$$

where $\mathbf{w}_h$ and $\mathbf{w}_v$ are horizontal and vertical feedforward weights and T is the transpose operator. The SC generates the trigger signal (m_t) that turns off the omnipause units to initiate saccades. The outputs of the horizontal and vertical burst generators are computed using saturating nonlinear functions (b_h and b_v)[8].

$$\frac{dE_h}{dt} = g \cdot b_h(m_h) \quad (5)$$

$$\frac{dE_v}{dt} = g \cdot b_v(m_v) \quad (6)$$

where E_h and E_v are horizontal and vertical eye position command signals from neural integrators (NIs), respectively and g ($0<g \leqq 1$) is a gain regulating the outputs of the burst generators. The eye position command signal and the output of the burst generator drive horizontal and vertical eye plants. Finally both position and velocity feedback signals are described by the following equations,

$$\mathbf{f}_p = fp \cdot \mathbf{r}_p \quad (7)$$

$$\tau_2 \frac{dfp}{dt} = -fp + k_1(E_h + E_v) \quad (8)$$

$$\mathbf{f}_v = fv \cdot \mathbf{r}_v \quad (9)$$

$$\tau_2 \frac{dfv}{dt} = -fv + k_2(\dot{E}_h + \dot{E}_v) \quad (10)$$

where $\mathbf{r}_p$ and $\mathbf{r}_v$ are position and velocity feedback weights and τ_2 is a small time constant. Both feedback signals are normalized by the scaling gains, k_1 and k_2.

3. Parameter optimization

We optimized the parameters in the 2-D saccadic model. The optimization procedure took two stages. First, the lateral interconnections (**W**) in the SC model were optimized using a hybrid method combining a genetic algorithm with a recurrent neural network training algorithm[9]. After obtaining optimized interconnections, we adjusted the horizontal and vertical feedforward weights ($\mathbf{w}_h$ and $\mathbf{w}_v$) and position and velocity feedback weights ($\mathbf{r}_p$ and $\mathbf{r}_v$) simultaneously so that the model produced accurate saccadic eye movements[10]. In the first step, the GA played an important role in finding better initial conditions for training the lateral weights. Once a candidate set of initial lateral weights had been selected by the GA, the recurrent neural network training algorithm was run to obtain a good fit between the output of the SC layer and actual discharge data from neurophysiological experiments[11]. After optimizing the lateral weights for 20 iterations, we evaluated the fitness function in the GA described by,

$$J_1(\mathbf{W}) = \frac{1}{2}\int_0^T \sum (y_i(t) - Y_i(t))^2 \quad (11)$$

where $Y_i(t)$ is actual SC visuomotor cell's discharge in time. In the second part of the optimization, the cost function is determined by,

$$J_2 = \frac{1}{2}\int_0^T (e_{hx} - e_{hd})^2 dt$$

$$+\frac{1}{2}\int_0^T (e_{va} - e_{vd})^2 dt \qquad (12)$$

where e_{ha} (e_{va}) and e_{hd} (e_{vd}) are horizontal (vertical) actual and desired eye trajectories in time. The derivatives of the cost function with respect to the parameters ($\mathbf{w}_h$, $\mathbf{w}_v$, $\mathbf{r}_p$, and $\mathbf{r}_v$) were computed in the feedback loop, and the parameters were updated using the gradient-descent method. For more details about the optimization procedure, see the references 9 and 10.

4. Simulation results

The model contains 30 x 30 units and generates oblique saccades in the range 2° to 20° in amplitude and 0° to 45° in direction. In the GA, each lateral weight W(d=0,1,2,...,7) was represented by 3-bit string, where d is the distance between two units in the SC model. Figure 2 shows one example of the genetic learning optimization, where the error indicates minimum residual calculated by the equation (11) at each generation. The proposed hybrid learning algorithm found a pattern of the lateral interconnections shown in Figure 3, which could generate realistic discharge of the SC cells. After fixing the lateral weights, we adjusted the feedforward and feedback weights. For simplicity, 15° and 20° target amplitudes and 0° and 30° directions were used. Figure 4 shows the error reduction during the learning process which simultaneously optimized the feedforward weights to the horizontal and vertical brainstem models and the feedback weights to the colliculus. Figure 5 shows an oblique saccade with a radial amplitude of 20° at 30° in direction. Once the model had been optimized to produce normal saccades, we tested its ability to also make perturbed movements which had not been included in the training set. To produce perturbed saccades, we turned the OPN model in the brainstem back on in the middle of a saccade, a process which interrupted saccades in midflight. As an emergent property, the model produced accurate saccades to the location of the visual input as shown in Figure 6, where the target was a pure horizontal 20° saccade.

5. Conclusion

The two-dimensional saccadic model was optimized using the hybrid training method. Our model was able to produce temporally perturbed saccades accurately as well as normal ones. This result supports the hypothesis that saccades are controlled by error feedback of signals that code efference copies of eye motion.

References

1. Keller, E. L. & Edelman, J. A. (1994). Use of the interrupted saccade paradigm to study spatial and temporal dynamics of saccadic burst cells in superior colliculus in monkey. *Journal of Neurophysiology*, 72, 2754-2770.
2. Keller, E. L., Gandhi, N. J., & Shieh, J. M. (1996). Endpoint accuracy in saccades interrupted by stimulation in the omnipause region in monkey. *Visual Neuroscience*, 13, 1059-1067.
3. Waitzman, D. A., Ma, T. P., Optican, L. M. & Wurtz, R. H. (1991). Superior colliculus mediates the dynamic characteristics of saccades. *Journal of Neurophysiology*, 66, 1716-1737.
4. Lefèvre, P. & Galiana, H. L. (1992). Dynamic feedback to the superior colliculus in a neural network model of the gaze control system. *Neural Networks*, 5, 871-900.
5. Optican, L. M. (1994). Control of saccade trajectory by the colliculus. In A. F. Fuchs, T. Brandt, U. Bhttner & D. S. Zee (Eds.), *Contemporary Ocular, Motor and Vestibular Research: A tribute to David A. Robinson* (pp. 98-105). Stuttgart: Georg Thieme.
6. Das, S., Keller, E. L. & Arai, K. (1996). A distributed model of the saccadic system: The effects of internal noise. *Neurocomputing*, 11, 245-269.
7. Arai, K., Das, S., Keller, E. L. & Aiyoshi, E. (1998). A distributed model of the saccadic system: Learning algorithms and simulations of temporally perturbed saccades. (submitted)
8. Van Gisbergen, J. A. M., Robinson, D. A. & Gielen S. (1981). A quantitative analysis of generation of saccadic eye movements by burst neurons. *Journal of Neurophysiology*, 45, 417-441.
9. Arai, K. & Aiyoshi, E. (1997). A hybrid learning algorithm integrating GA with neural networks for saccade generation model (in Japanese). The trans. of the IEE of Japan, 117-C, 150-157.
10. Arai, K. & Aiyoshi, E. (1998). A learning algorithm for saccade model with distributed feedback mechanism (in Japanese). The trans. of the IEE of Japan, 118-C, 685-693.
11. Anderson, R. W., Keller E.L., Gandhi, N. J., & Das, S. (1998). Two-dimensional saccade-related population activity in superior colliculus in monkey. *Journal of Neurophysiology*. In press.

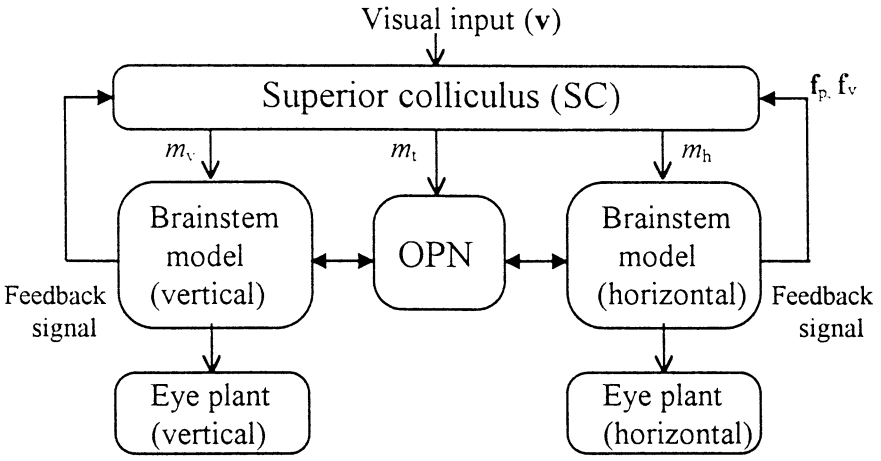


Figure 1. 2-D distributed saccadic model

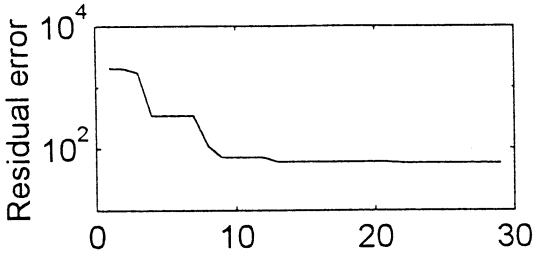


Figure 2 GA learning process

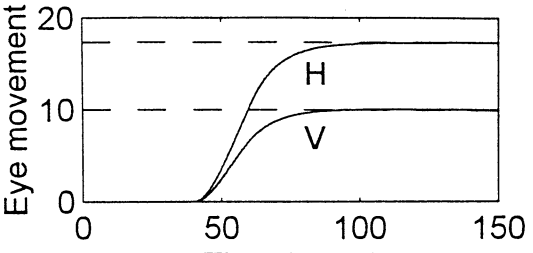


Figure 5 Oblique saccades

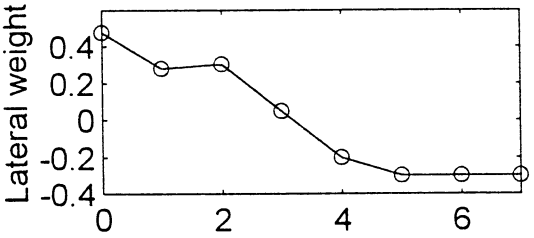


Figure 3 Lateral weight

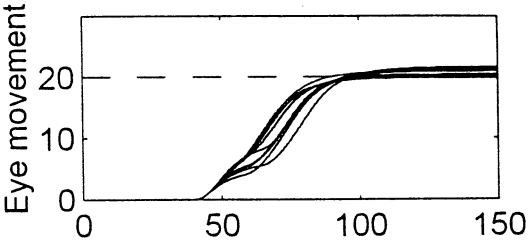


Figure 6 Interrupted saccades

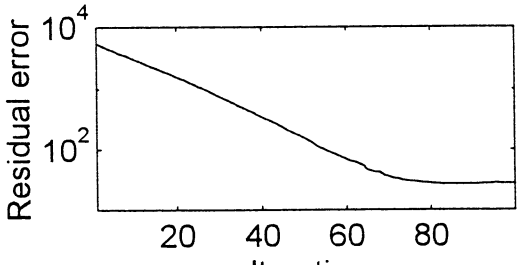


Figure 4 Learning process

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Reinforcement Learning Approach to Cooperative Carrying Problem

Ken-ichiroh Kawakami, Kazuhiro Ohkura, Kanji Ueda

Department of Mechanical Engineering,
Kobe University, Rokko-dai, Nada-ku, Kobe 657, Japan.
E-mail: kawakami{ohkura, ueda}@mi-2.mech.kobe-u.ac.jp

Key Words: Cooperative Behavior , Reinforcement Learning , Cooperative Carrying Problem, Multi-Agent System

Abstract

This paper shows an approach to behavior acquisition to a cooperative problem where two robots are connected to each other. Each robot has its own movement controller. To compensate the dynamics in the environment, a prediction mechanism for the future state of the other robot is introduced. Computer simulations are conducted to evaluate effectiveness of proposed mechanisms.

1. Introduction

A Distributed Autonomous System (DAS) has been considered. It is effective for the so-called Bottle neck Problem of the computing system theory. DAS can decompose a given task into several sub-tasks which are solved by sub-systems of the DAS, and it may bring a high response speed to the system. Such a high speed response system is necessary for a dynamic environment, such as manufacturing systems for a diversifier. This paper proposes a method for designing such a DAS. As a basic study for DAS, we will give a solution for a Multi-Agent Learning Problem, which is an application of DAS. We take up Cooperative Carrying Problem (CCP) as one of the Multi-Agent Learning Problems, and show a reinforcement learning approach to the CCP.

2. The Cooperative Carrying Problem

The Cooperative Carrying Problem (CCP) is one of the cooperation problems among autonomous agents (or robots). In this paper, we try to solve the CCP by two robots (Fig.1). The CCP is described as follows. Two autonomous robots are connected to each other by a bar-like load. The joints between the load and the robots can rotate freely. Every robot has eight proximity sensors and two driving wheels. Differences between the headed direction can be calculated from the joint angles.

Generally speaking, the CCP can be classified as a non-holonomic problem in the field of the control engineering, to which conventional feedback control methods cannot be applied. To date, a number of approaches have been applied to this problem, but they usually assign different roles to the robots, such as a leader and a follower. One robot finds an appropriate route to the goal, and the other robot follows the leader without disturbing the leader's movement.

In the present study, the authors try to solve this problem by a reinforcement learning method. Assumptions are as follows:
(1) Both robots define their actions and learn how to behave independently.
(2) The robots have the same control system and same reinforcement learning architectures.
(3) In the beginning, they have no specific rules for aviod an obstacle, approaching the goal, and cooperating with each other.

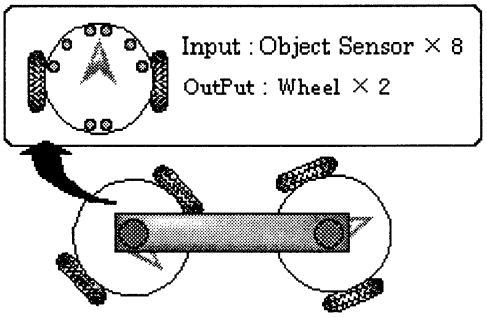


Fig.1 Cooperative Carrying Problem by two robots

3.1 Approach

The reinforcement learning technique is ensured to converge in a static environment. In the CCP, however, taking movement of the other agent into account makes the state-space dynamic. Therefore, we cannot directly apply reinforcement learning to the Multi-Agent Reinforment Learning Problem.

In a dynamic environment, the reinforcement learning methods have two difficulties. One is incomplete recognition of situation, and the other is uncertain transition of situation. For example, Fig.2 shows a example of avoiding a moving object. The robot at time t cannot recognize the state of the object at time $t+1$ and it cannot know transition of situation, because its sensory inputs at

time t (Fig.2a) are the same whether the object will moves like in Fig.2b or in Fig.2c. So, if the robot selects same action in the same situation, its feasibility depends on the motion of the object.

In order to make the robots behave appropriately, we introduce a method that uses an extra learning mechanism for predicting the state of the other robot. Each robot has to have a "static" information of the other's future state (time t+1 in Fig.3), since it is the future state that militates for a result of its action when the action will be evaluated. If each robot could precisely predict the other's future state, it would not need to consides the other's movement as a "dynamic" information. In this case, we can regard the robots' "dynamic" learning-space as "static" learning-space. This prediction mechanism (called Predictor) is constructed by means of the Stochastic Learning Automaton (SLA).

The main learning mechanism, that selects the robot's actions according to inputs, is the Instance-Based Classifier Generator. The rule-string consists of the sensory inputs, output of the predictor and the robot's action (Fig.4).

3.2 Instance-Based Classifier Generator

Instance-Based Classifier Generator (IBCG) (Nakamura, 1997) is one of the reinforcement learning methods that has been developed as a classifier system. Its overview is as follows.

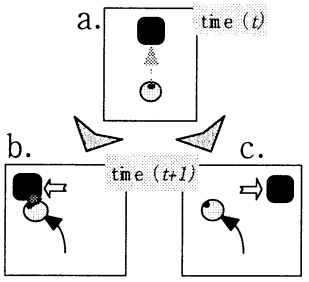


Fig.2 Dynamic environment

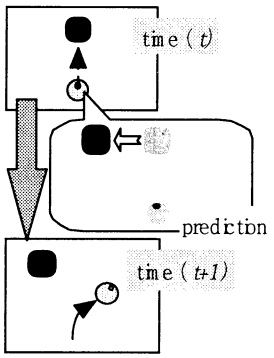


Fig.3 Prediction

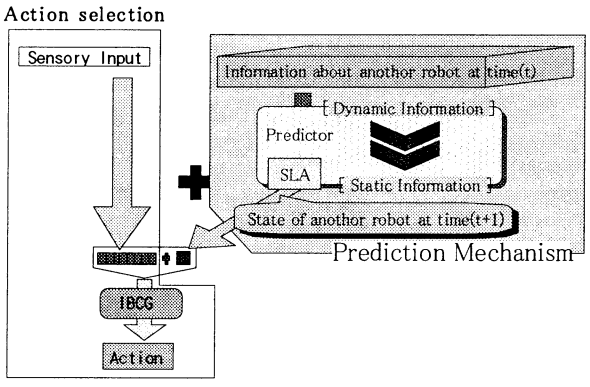


Fig.4 Learninig Structure

In order to aquire new behaviors with the use of the abovementioned merits of reinforcement learning, a learner needs to explore its wide state space and to continue to learn even if the delay of reward is long.

IBCG adopts two distinctive mechanisms to deal with the wide state space. One is the instance-based generation of action-rules. A robot with IBCG explores its environment while determining actions in response to the perceived unknown states. The IBCG memorizes only the actually experienced state-action pair as an action-rule. The other mechanism is introduction of generalized rules, which cover multiple regions in the state space. In contrast to the genetic algorithms, a new generalized rule is reproduced from only one parent. The generated rules complete with one another, and only the rules having high utility survive. Utility estimates usefulness of the rule for achieving a task. We expect that the survivor would have acquired some useful behaviors.

In addition, IBCG adopts a new scheme of a set of assignment to cope with the long delay of reward.

The IBCG works effectively when significiant state-space of learning is smaller than the entire state-space, because IBCG makes rules only for the actually experienced states and memorize only significiant rules.

3.3 Predictor

SLA which constructs the Predictor at the t-th time step is expressed as follows:

$$SLA = \{\Phi, P_{sta}(t+1), Inp, A, G\} \quad (1)$$

where Φ is the state of the SLA, Inp is a history the input for the SLA.

$$Inp = \{sta(t), sta(t-1), ..., sta(t-n)\} \quad (2)$$

$sta(t)$ is the state of the other robot at the t-th time step. Here, input for the SLA is the history of the states of the other robot, and n=3 in our experiments.

In our SLA, the state of the SLA does not include the output.

G is an identity mapping, and A is an updating algorithm $L_{R\text{-}P}$-scheme. $P_{sta}(t+1)$ is the output of the SLA. It is a predicted state of the other robot at the t+1-th time-step.

$$\Phi = Inp \quad (3)$$

If $P_{sta}(t+1)$ equals to the actual state of the other robot at the t+1-th time step, then the au-

tomaton is positively reinforced. Otherwise, it is negatively reinforced.

In the CCP, we use the differences of the headed direction between the two robots as a state of the other (Fig.6 ψ). $P_{sta}(t+1)$ is the differences of the headed direction between the two robots at the t+1-th time step, and $sta(t)$ is the differences of the headed direction between the two robots at the t-th time step.

4.1 Simulation

Fig.5 shows robots in our computer simulations. We try to solve the CCP by computer simulation in two kinds of environment. One is shown in Fig.7 (case 1), and another is shown in Fig.8 (case 2). According to the inputs from the sensors and the predictors, the robots select their actions from five basic actions: Move Forward, Turn Right, Turn Left, Rotate Right, and Rotate Left.

The task is to move round the square-shaped region in the clockwise direction. The robots are given sequential rewards when they move in the clockwise direction. They are given penalty when they move in the counter-clockwise direction and when they collide against the wall or the object in the center of the environment.

4.2 Results

Case 1:

Fig.9 shows traces of the robots in case 1. The robots move cooperatively and stably around the object in the clockwise direction.

The robot that is closer to the object (called robot A) aquires such a behavior that it moves with sensing the object in its right side. Rules for this behavior are (1) to move forward when there are inputs only to the right-side sensors, and (2) to turn right when there are inputs to the front sensors.

The robot that is farther from the object (called robot B) aquires a behavior that it takes after the other robot's motion. Its action rules are (1) to move forward as long as its headed direction is equal to that of the other robot's headed direction, (2) to turn right when robot A starts to turn right, and (3) to turn left when robot A starts to turn left. It should be noted that the other robot's direction considered here is a predicted state by the predictor.

The two robots acquire different behaviors. Robot A aquires behavior of moving round a square-shaped region in the clockwise direction without considering the other robot's motion. Robot B aquire behavior of cooperative movement by consideration take only after the other robot's motion. That is to say, robot A learns to become a leader, and robot B learns to become a follower. Thus, we can see an autonomous specialization of the roles of the leader and the follower, and it makes the robots behave cooperatively to execute their task.

Case 2:

Fig.10 shows traces of the robots in case 2. As shown in case 1, the robots aquire two kinds of behavior. One is moving along the wall as a leader, the other one is taking after the other robot's motion as a follower. In contrast to the case 1, those roles are not stationary but alternate depending on their situation.

Their behaviors are as follows. (1) They move along the wall as long as neither of them sense the wall in their forward direction (In area A and C in Fig.10). (2) When only one of them finds a wall in front, it begins to turn to right and the other robot responds by turning quickly to the same direction simultaneously (the robot A in area B and the robot B in area D in Fig.10). (3) A robot which finds the wall on its left-side goes along the wall, and the other robot traces the movement (the robot B in area B and the robot A in area D in Fig.10).

While in the case of behavior (1), the robots are coordinative. In the case of behavior (2) and (3), the robot that find the wall first becomes the "leader", and the other becomes the "follower". In area B in Fig.10, robot A is the leader and robot B is the follower. In area D in Fig.10, robot B is the leader and robot A is the follower. In area A and C in Fig.10, both robots coordinate and they behave following the same sets of the rules. Both robots

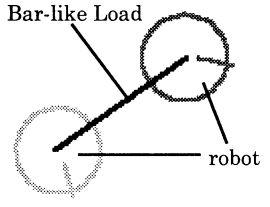


Fig.5 Robots in Simulator

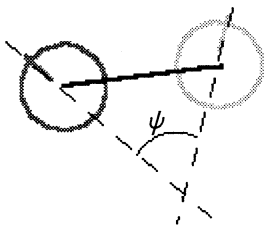


Fig.6 Prediction State

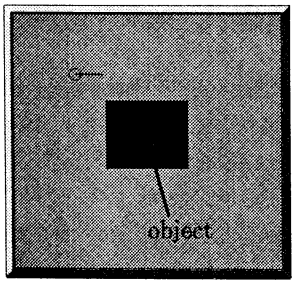
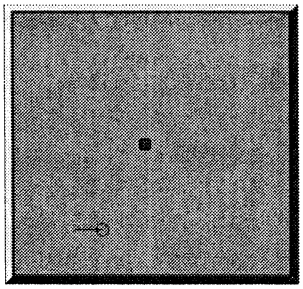


Fig.7 Environment 1


Fig.8 Environment 2

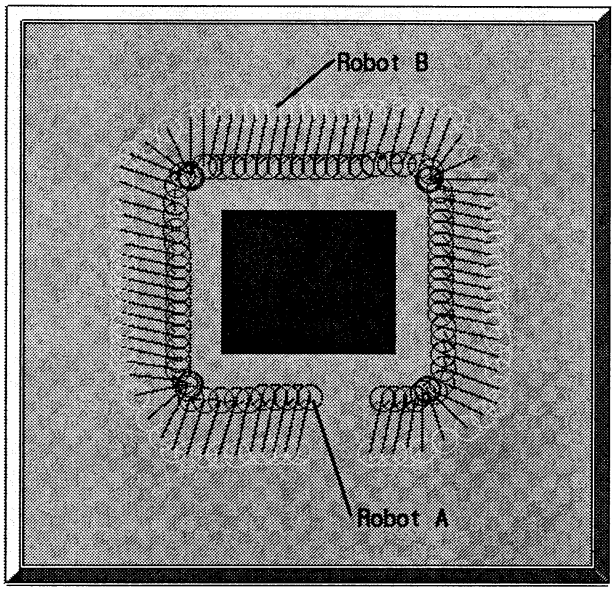


Fig.9 Trace of behavior in case 1

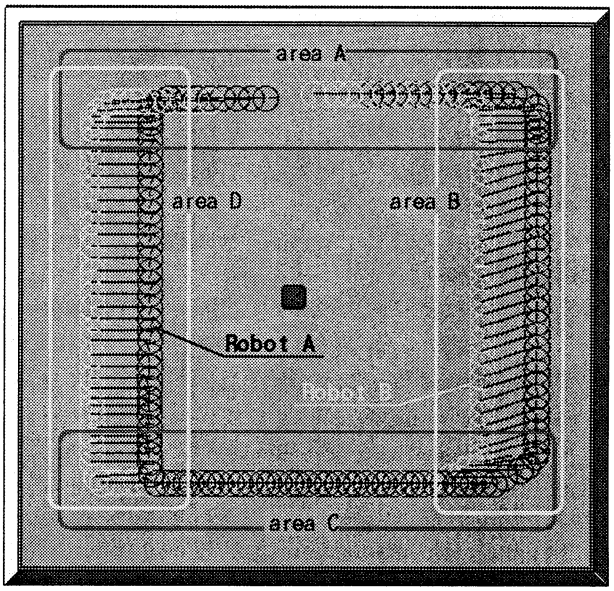


Fig.10 Trace of behavior in case 2

learn these avoidance behavior (leader) and following behavior (follower). They exchange their roles of leader and follower depending on their sensory inputs. Therefore, it can be concluded that a specialization of a leader or a follower emerges between the homogeneous robots.

The results are summarized as follows: Different sets of action rules are used when the two robots behave as a "leader" and a "follower". The leader's rules are,

(1) Moving forward as long as there is no object in front.

(2) Turning right when there is no object in right side (in case 1).

Turning right when there is a wall in front (in case 2).

The follower's rules are,

(1) moving forward when its headed direction is equals to the other's.

(2) Turning right when the other turn right.

(3) Turning left when the other turn left.

At the results, the robots execute their tasks following about ten rules; the number of rules can be 200 at the maximum.

5. Conclusion

We have introduced a method using an extra learning mechanism that predicts the other robot's state in order to solve the multi-agent learning problem in the dynamic environment. We verified the efficiency of the method by the CCP through computer simulations. The results of the experiments show that the robots can efficiently acquire cooperative behavior.

Acknowledgement

This research is partially done under the financial support through "Methodology for Emergent Synthesis" project (project number: 96P00702) in Research for the Future Program of the Japan Society for the Promotion of Science (JSPS). The authors acknowledge that the discussion with other project members was directive for this research.

References

[1] L. P. Kaelbling, M. L. Littman, "Reinforcement Learning: A Survey", Journal of Artificial Intelligence Research 4, pp. 237-285, 1996.

[2] A.Arai, K.Miyazaki, and S.Kobayashi, "Methodology in Multi-Agent Reinforcement Learning --Approachs by Q-Learning and Profit Sharing" , Journal of Japanese Society for Artificial Intelligence, vol.13 No.4, 1998, pp.105-114. (in Japanese)

[3] K.Narendra, and M.A.L.Thathachar, Learning Automata -- An Introduction, Prentice-Hall International Editions, 1991.

[4] Y.Nakamura, S.Ohnishi, K.Ohkura, and K.Ueda, "Instance-Based Reinforcement Learning for Robot Path Finding in Continuous Space", in Proc. IEEE Int. Conference on System, Man, and Cybernetics, pp. 1228-1234, 1997.

[5] K.Kosuge, T.Oosumi, and K.Chib, "Decentralized Control of Multiple Mobile Robots Handling a Single Object in Coordination", Journal of the Robotics Society of Japan, vol.16, No.1, pp.87-95, 1998. (in Japanese)

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

On the Emergence of Motion Patterns in Locomotion Systems

M.M. Svinin K. Yamada K. Ueda

Department of Mechanical Engineering, Kobe University
Rokkodai, Nada-ku, Kobe 657–8501, Japan

Abstract

Emergence of motion patterns in locomotion systems is studied in this paper. A classifier system is used to generate control commands. The system is applied to the gait acquisition for an eight leg robot. It is shown under simulation that periodical gaits emerge as a result of interaction between the classifier and the robot mechanical system.

1 Introduction

Emergence of sensible motions in living organisms and robots is an interesting phenomenon. The phenomenon is very complex, especially for locomotions where legs interact with the ground and demonstrate a cooperative behaviour. The motion emergence mechanism is not understood yet, and it challenges researchers and engineers working in the field of robotics.

In locomotion systems, we are normally interested in the emergence of stable periodical gaits. Classical control approaches [1] gives us only a basic framework for studying mechanics of locomotion. However, their use is limited, especially if autonomy of the robot is required to function in unknown environments.

In living organisms, gaits are the product of the interaction between the nervous and muscular systems. While the muscular system can be well formalized within theoretical mechanics, modeling of the nervous system is still an open field of research. One possible approach, using network of neural oscillators for evolving periodical gaits, has been reported in literature [2, 3]. Another approach, based on the ideas of complex adaptive systems [4], is presented in this paper. In our previous research [5], we have developed a classifier system and applied it to controlling a simple two-leg robot supported by casters. In this paper we test it for a more complex example of an eight-leg robot.

This paper is organized as follows. We briefly describe the structure of the classifier system in Section 2, and Section 3 presents simulation results.

2 Structure of the Classifier System

Let n_s be the number of sensors a robot is equipped with, and $X = \{x_1, \ldots, x_{n_s}\}^{\mathrm{T}}$ is the sensory input of the robot. No explicit knowledge on the robot internal model is available in the control system.

The system operates on a set of action rules, R. The rule $r \in R$ is defined as follows: $r :=< V, W, u, a >$, where $V = \{v_1, \ldots, v_{n_s}\}^{\mathrm{T}}$ is the state vector associated with and memorized in the rule r, $W = \{w_1, \ldots, w_{n_s}\}^{\mathrm{T}}$ is the weight vector, u is the utility of the rule, a is the action corresponding to the rule r.

If V matches in some sense the current sensory input X, the rule r becomes active and can trigger its action a. The weight vector W is used for comparing V and X. We set $w_i \in [0, 1]$. The closer w_i to zero, the less important the measurement of the ith sensor. The rules for which $W = 0$ are called indefinite. These rules can be activated anywhere regardless of the current state X the robot is in. All the other rules are definite. They can be activated in a vicinity of V, and the vicinity is defined using the weight W.

The rule's specificity $\lambda = \frac{1}{n_s}\sum_{i=1}^{n_s} w_i$ serves as a measure of definiteness of the rule. In the beginning R consist of only one indefinite rule with initially assigned utility u_0. As learning progresses, the total number of rules in R, n_r, varies by reproduction and extinction.

2.1 Action selection

The rules in R compete with each other for the right to trigger their actions. The selection procedure is relatively simple. First, all the rules $r_j \in R$, $j = 1, \ldots, n_r$, compare their own state vector $V^j = \{v_1^j, \ldots, v_{n_s}^j\}^{\mathrm{T}}$ with current sensory state X. The normalized weighted distance is defined as: $\sigma_j^2 = \sum_{k=1}^{n_s}(\frac{x_k - v_k^j}{d_k} w_k^j)^2/n_s$, where d_k is a time-dependent scaling parameter.

The winner rule is selected with the probability given by the weighted Boltzmann distribution: $P(r_j) = m_j\lambda_j \exp(u_j/T)/\sum_{k=1}^{n_r} m_k\lambda_k \exp(u_k/T)$,

where the matching rate $m_j = \exp(-T_m \sigma_j^2)$, and T_m is a constant.

2.2 Temporal credit assignment

The utilities of the rules are updated every time after the winner executes its action. The mechanism of utility adjustment consists of four parts.

1. Direct payoff distribution The direct payoff P is given to the winner rules only in specific states. There are two types of payoff: reward ($P > 0$) and punishment ($P < 0$). The payoff is spreading back along the sequence of the rules triggered their actions (i.e., to the current and previous winners) with the discount rate $\gamma > 0$: $u_w^{(k)} \leftarrow u_w^{(k)} + \gamma^k P, \quad k = 0, \ldots, N$, where N is the depth of the winners chain. Here, we denote the parent of r_w as $r_w^{(1)}$, the parent of $r_w^{(1)}$ as $r_w^{(2)}$ and so on.

2. "Bucket brigade" strategy. The current winner r_w hands over a part of its utility, Δu, back to the previous winner, $r_w^{(1)}$: $u_w^{(1)} \leftarrow u_w^{(1)} + \Delta u$, where $\Delta u = \kappa (u_w - u_w^{(1)}), \kappa \in [0, 1]$, if $u_w > u_w^{(1)}$ and $\Delta u = 0$ otherwise.

3. Taxation. Whenever a definite rule r_w triggers its action, its utility is updated as $u_w \leftarrow (1 - c_f)\, u_w$.

4. Evaporation. All the rules reduce their utilities at the evaporation rate $\eta < 1$ when the robot reaches the goal state: $u_w \leftarrow \eta\, u_w$. The rules decreasing their utility below the threshold $u^{\min}$ are removed.

2.3 Reproduction of rules

The winner rule r_w always generates a new rule r_c except for the case when the action triggered by r_w has led to collision with the external objects. If the winner is the indefinite rule, the reproduced rule parameters are set as $v_i^c = x_i$, $w_i^c = 1$, $i = 1, \ldots, n_s$, $a_c = a_w$, $u_c = u_w$.

If the winner is a definite rule, the newly produced rules are called generalized. The winner reproduces a generalized rule r_c provided that its matching rate m_w is within a certain reproduction threshold θ_r, i.e., $m_w < \theta_r$, where $\theta_r = \exp(-T_r u_w)$ and T_r is a constant. The vectors V^c and W^c for the new generalized rule are set as follows: $v_i^c = x_i$, $w_i^c = 1 - \frac{|x_i - v_i^c|}{d_i}$, $i = 1, \ldots, n_s$, $a_c = a_w$, $u_c = \lambda_c u_w$.

3 Gait Acquisition

The system described in the previous section is applied to acquisition of gaits of an eight-legs locomotion robot, named OCT1-b and produced by AAAI Ltd.



Figure 1: OCT1-b robot

Table 1: List of actions for the four front legs

No.	Description	q_1	q_2
1	Move forward and lift down	$+39°$	$-10°$
2	Move back and lift down	$-7°$	$-10°$
3	Move back and lift up	$-7°$	$+35°$
4	Move forward and lift up	$+39°$	$+35°$

Table 2: List of actions for the four rear legs

No.	Description	q_1	q_2
1	Move forward and lift down	$+7°$	$-10°$
2	Move back and lift down	$-39°$	$-10°$
3	Move back and lift up	$-39°$	$+35°$
4	Move forward and lift up	$+7°$	$+35°$

The robot is shown in Fig. 1. Every leg has two degree of mobility (q_1 for the rotation in the body plane, and q_2 for lifting up and down), and is equipped with two angle sensors. In addition, there are three light sensors at front part of the robot body[1]. Thus, the dimension of the sensor space is $n_s = 11$.

One classifier system is used for controlling the robot. Based on the current sensory input, the control system outputs the commanded action for each leg. There are four actions for every leg, so the total dimension of the action space ($4^8 = 65536$) is rather large. The actions for the first four front and rear legs are listed in Table 1 and Table 2. The actions as well as the values of the commanded angles are taken from the factory settings for the sample motions of the OCT1-b robot.

The actions generated by the control system can lead to stable or unstable configurations of the robot as shown in Fig. 2. The stability is understood in the static sense, i.e., it is preserved if the gravity center of the body, projected onto the ground, lies within

[1]There are also touch sensors and infrared sensors in the full configuration of the OCT1-b robot. However, they were not used in the simulation

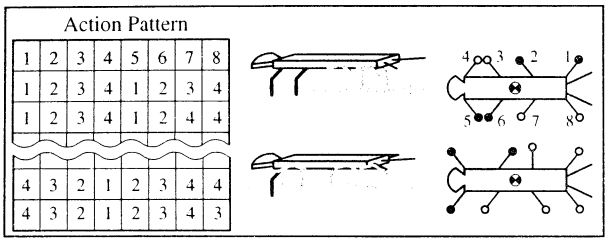


Figure 2: Actions and configurations

the contact leg polygon. Whenever the robot comes into an unstable configuration, the rule that led to this configuration, is punished.

3.1 Task and parameter settings

In the computer experiments described below, we pursuing a simple goal: to make the robot able to reach a light source located at the distance of 6 lengths of the robot body. The teaching task is decomposed into three consequent subtasks: 1) To make the robot able to walk stably. 2) To make the robot able to move straight forward. 3) To make the robot able to reach the goal.

In the 1st subtask, the robot is rewarded for any stably motion, and is punished whenever it falls down. In the 2nd subtask, having acquired the ability to move stably, the robot is rewarded for moving forward, and is punished for stepping back and for falling down. The same rewarding scheme is accepted for the 3rd task, except there is an additional reward for reaching the light source. In what follows, this reward is called global while all the other reward are called local.

The time interval between the starting motion and stopping the robot is called an episode. The episodes are updated when the robot reaches the light source, or when the number of produced actions reaches 500. Note that the robot does not have *a priori* knowledge of the environment and the goal coordinates.

The direct payoff $P = 10$ is set for the global reward, and $P = -2\%$ of the corresponding utility is set for punishment. As the local reward, $P = 0.1$ is set for the 1st subtask, and $P = 1.0$ is set for the 2nd subtask. The maximum number of rules is restricted to $n_r = 500$. When reproduction exceeds the limit, a definite rule with the minimum utility is removed. The parameters are set as follows: $T = 3$, $T_r = 0.5$, $T_m = 100$, $c_f = 0.001$, $\gamma = 0.8$, $\eta = 0.95$, $\kappa = 0.1$, $u_{\min} = 9.5$, $u_0 = 10$.

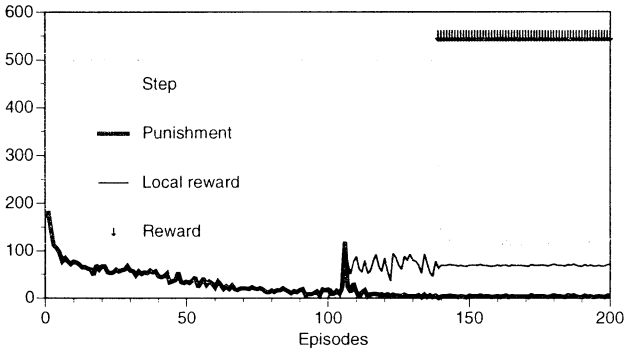


Figure 3: Learning history.

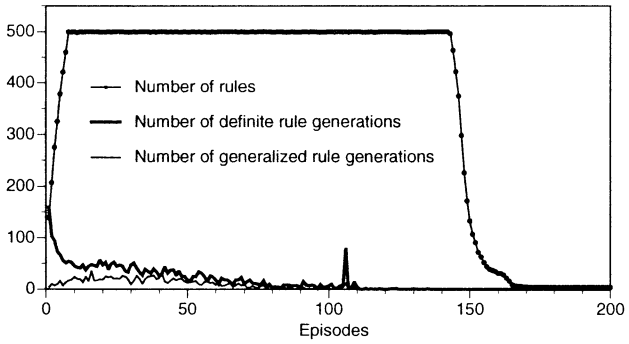


Figure 4: Dynamics of rule population.

3.2 Simulation Results

Feasibility of the control schemes is tested under simulation. The switches from the 1st subtask to the 2nd, and from 2nd to 3rd happen, respectively, during the 100th and the 130th episodes.

The numbers of local rewards and punishments are plotted in Fig. 3. Here, we also indicate the episodes at which the robot obtains the global reward. For the 1st subtask, contribution of the local rewards is not significant The learning rate is defined mainly by the punishments, which are gradually decrease. There is a splash of punishment when the robot switches from the 1st to the 2nd subtask. Note that in the 1st and 2nd subtask the robot takes the maximum number of steps since the global goal is not fixed there.

Fig. 4 plots the total number of rules, and, respectively, the number of generations of the definite and the generalized rules. As soon as the global goal is fixed, the total number of rules gradually decreases. The comparison of Fig. 4 and Fig. 3 reveals a high correlation between the total number of rules and the total number of steps necessary to reach the global goal.

As can be seen, there also exists a high correlation between the number of punishments and the number of generation of the new rules. This indicates, indirectly,

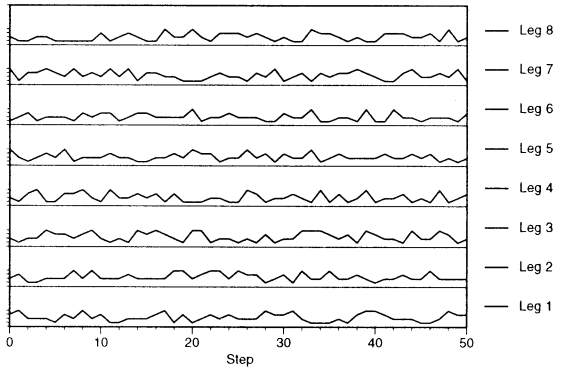


Figure 5: Gaits in the 50th episode

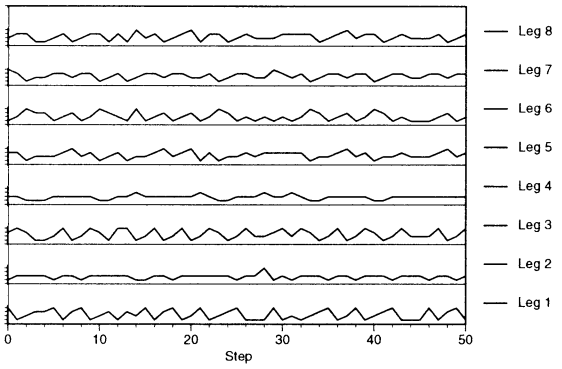


Figure 6: Gaits in the 130th episode

that the exploration function of the indefinite rule is gradually subsumed by the exploitation function of the definite rules with high utilities. In fact, only particular definite rules, causing "useful" behavior, trigger their actions and increase their utility. Conversely, the "irrelevant" rules decrease their utility and evaporate eventually. Thus, after some time the behavior acquired by the survivors becomes dominant.

Finally, the robot gaits, acquired respectively at the 50th, 130th, and 200th episodes, are shown in Fig. 5, 6, and 7. Here, for every leg we plot the action numbers taken at the corresponding steps. The results demonstrate that as learning progresses, the acquired gaits become periodic. It is interesting to note that the rate of convergence to the periodic motion depends on the presence of the global goal.

4 Summary

Emergence of motion patterns in locomotion robots has been studied in this paper. A classifier system, generating the robot actions, was used as model of

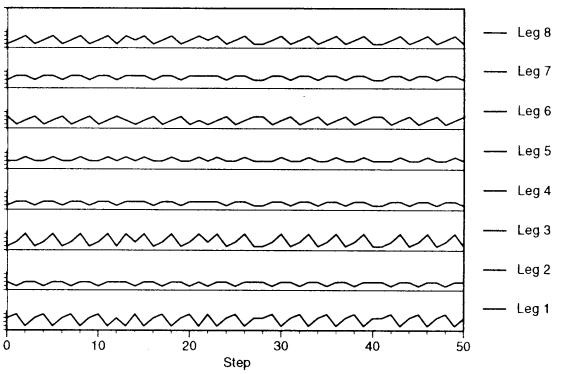


Figure 7: Gaits in the 200th episode.

the control system. The system has been applied to the gait acquisition for an eight leg walking robot. A special teaching strategy, has been tested. The strategy realizes step-by-step learning of stable walking, straight-forward moving, and reaching a goal.

The classifier system and the teaching stratefy have been tested under simulation. It has been demonstrated that periodical gaits can emerge as a result of interaction between the classifier and the robot mechanical system. Our initial results show feasibility of the control approach. In future, we plan to test the approach under experiments with real OCT-1 walking robot, and to extend this approach to different schemes of local inter-leg coordination.

References

[1] Vukobratovic, M., *Walking Robots and Anthropomorphic Mechanisms*, Berlin, Springer Verlag, 1976.

[2] Taga G., Yamaguchi Y., and Shimizu S., "Self-Organized Control of Bipedal Locomotion by Neural Oscillators in Unpredictable Environment," in *Biological Cybernetics*, Vol. 65, 1991, pp. 147–159.

[3] Katayama O., Maeda T., Uno M., Kurematsu Y., and Kitamura S., "Neuro Oscillator for the Motion Generation of Biped Locomotive Robots," *Proc. Japan-USA Symp. on Flexible Automation*, Nantes, France, 1994, Vol. 2, pp. 477–480.

[4] Holland, J.H., *Hidden Order. How Adaptation Builds Complexity*, New York, Addison-Wesley, 1995.

[5] Kuroyama, K., Svinin, M.M., and Ueda, K., "A classifier system for reinforcement learning control of autonomous robots," in *Intelligent Autonomous Systems*, Y. Kakazu, M. Wada, and T. Sato (Eds.) *Proc. 5th Int. Conf. IAS-5*, June 1–4, 1998, Sapporo, Japan, IOS Press, Amsterdam, pp. 304–311.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Towards man-machine creativity in conceptual design

Kryssanov, V.V. Tamaki, H., and Kitamura, S.
Faculty of Engineering, Kobe University, Rokkodai, Nada, Kobe 657-8501, Japan
E-mail: kryssanov@ziong.cs.kobe-u.ac.jp

Abstract

This paper discusses how human creativity could be enhanced with computers in the field of engineering design. We address this issue by starting from a description of a cognitive image of the creative reasoning process. We then argue that this image is directly applicable to engineering design, and that the understanding of the mechanisms of creativity could lay foundation for a computer-based design methodology. A formal model of the design creative process is introduced, and the model essential properties are described. A synthesis methodology for conceptual design is suggested, and possible directions and approaches for future studies of the design creative process and its computational treatment are given.

Key words: *Engineering design, creativity, synthesis.*

1. Introduction

Traditionally, engineering design is a prominent field for the application of computers and information technologies. However, computer tools capable of supporting designers in the early creative phase of design are still rare. In contrast to such areas of human activity as art, science, and psychology (e.g., see [1]), there is little theoretical research on creativity in engineering design. At the same time, recent investigations in the cognitive sciences have shown that computers might significantly enhance the human creative potential [2, 3].

The aim of the presented research is to explore the mechanisms of creativity in conceptual design. This paper is also to identify possible research directions for future studies of the design creative process and its computational treatment, issues that have so far not been discussed in the literature to a considerable extent.

We approach creativity from the standpoints of psychology, decision-making, and computer sciences. In the next section, we discuss a classical model of the creative process in the context of the design process. Cognitive mechanisms behind human creativity are then outlined. In Section 3, we introduce a model of the creative process based on the design synthesis-analysis interactions. A computational model of creative reasoning is presented in Section 4. In this section, we also investigate some formal properties of the model. Finally, in Section 5, the paper conclusions are given.

2. On creativity

A classical model of the creative process that has been taken for the basis in many studies on the subject is the simple cognitive pattern suggested by Graham Wallis many years ago [4]. Figure 1 gives an interpretation of this model. The creative process is divided into the following four stages:

1. Preparation: specifying the problem and gathering the relevant and potentially useful information;
2. Incubation: step back from the problem, contemplate, and work through the problem;
3. Illumination: ideas arise from the mind to provide for the problem solution;
4. Verification: the ideas are checked whether or not they meet the problem specification.

The creative process has a complex, iterative dynamics that manifests itself on two major levels: (i) the system level and (ii) the domain level. The former gives a methodology (or structure, style, genre), while the latter determines the application area (or conceptual content). There are two major cognitive mechanisms of creative thought: divergent thinking that is the ability to generate original, distinct, and elaborate ideas, and convergent thinking that is the ability to logically evaluate and choose the best solution from a variety of alternatives. It is understood that the first mechanism plays the prime role during the incubation and illumination stages, and the second – during the preparation and verification stages. Among the ways of achieving a creative solution, the following three are the most general: 1) serendipity, 2) similarity, and 3) meditation. Creativity is classified into real-time creativity that requires solution in a short interval of time and multistage creativity that assumes sufficient time for the solution generation. There are many other

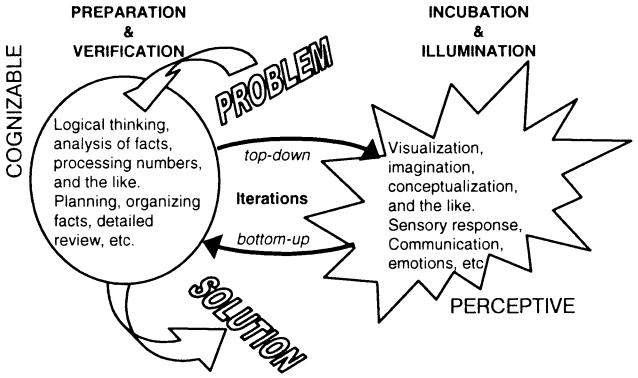


Figure1. The creative process.

aspects of creativity (for details, see [4]), but from the standpoint of design, the most pragmatically interesting definition of creativity includes the ability to take existing objects and combine them in different ways for new purposes. As we shall then see, the cognitive creative process can be projected and interpreted in terms of the design process, when each of the creative stages is given a parallel in the synthesis-analysis design cycle. Focusing on real-time creativity (that naturally and often takes place in engineering design) and considering the similarity-based approach to obtaining creative solutions, let us discuss the latter assertion more formally.

3. Design creative process model

A promising way to provide guidance in the creation of computer-based design systems capable of enhancing human creativity is through the development of a formal model of the design creative process. For this purpose, we will use the mathematical apparatus of algebraic specification [5].

A design can generally be defined in terms of a sign system $\Xi = \langle S, V, C, R, A\rangle$, where S is a set of sorts for signs in Ξ, V is a set of data sorts, R is a set of relations and functions defined on signs, A is a set of axioms constraining the possible signs. All the sorts are partially ordered by sort-subsort relation (denoted $\leq$), and there is a level structure imposed on the ordering so that data sorts are lower than sign sorts and there is a unique sort (top sort) at the highest level. C is a set of functions defined as constructors to build level n signs from signs at levels n or less. There is a priority partial ordering on each level constructors. (See [6] the original work on sign systems.)

By the above definition, Ξ can be seen as a design formal theory, where S, V, and some initial C come from a general domain theory, A determines design requirements, and R gives a context for the design. We can now define the design creative process as follows:

$\Xi_0 \Rightarrow \Xi_1 \Rightarrow \ldots \Rightarrow \Xi_{k-1} \Rightarrow \Omega = \langle \Sigma \cup \Phi, \Lambda\rangle$,

where $\Rightarrow$ denotes a morphism (partial mapping) from a sign system Ξ_i to Ξ_{i+1}, Ω is a completed design model, Σ is an order-sorted signature, Φ is a finite set of (closed) logical formulas written in terms of Σ, Λ is an order-sorted algebra of the signature Σ such that all the formulas entering Φ are true.

A morphism consists of partial functions (all denoted $\mathcal{M}$) which map sorts, constructors, predicates and functions of Ξ_i to sorts, constructors, predicates and functions of Ξ_{i+1} respectively, in such a way that

if $s \leq s'$, $s, s' \in S$, then $\mathcal{M}(s) \leq \mathcal{M}(s')$;

if c: $s_1 \ldots s_n \rightarrow s$, $s_1 \ldots s_n \in S$, is a constructor (or function) of Ξ_i, then (if defined) $\mathcal{M}(c)$: $\mathcal{M}(s_1) \ldots \mathcal{M}(s_n) \rightarrow \mathcal{M}(s)$ is a constructor (or function) of Ξ_{i+1};

if p: $s_1 \ldots s_n$ is a predicate of Ξ_i, then (if defined) $\mathcal{M}(p)$: $\mathcal{M}(s_1) \ldots \mathcal{M}(s_n)$ is a predicate of Ξ_{i+1}; and

$\mathcal{M}$ is the identity on all the data sorts and operations in Ξ_i.

Each of the sign systems Ξ_i is to specify a design state, and it directly corresponds to a representation of that which is called in the cognitive sciences a conceptual space. The design process is seen as a progressive but non-monotonic transition from a state described in Ξ_0 as an initial specification to a design model Ω that is composed of $\Sigma \cup \Phi$ – a completed design requirement specification (a formal theory) and Λ its realization (a model of the theory). A morphism $\mathcal{M}$ between two sign systems characterizes a transition from one design stable state to another stable state, where a stable state corresponds to a (partially) closed design model, i.e. a model that does not require any external (in respect to the design space) data to determine the further feasible evolution of the model. We will distinguish up-coming transitions among all the possible transitions during designing. An up-coming transition is initiated by closing a synthesis-analysis cycle. This cycle generally has an iterative character, where each iteration starts from identification of one or more design solutions consistent with the existing design requirements and ends by evaluation of the obtained design description for conformance with expected performances.

The design synthesis process results in elaboration of the design by adding new elements to S, V, and/or C, whereas the analysis process leads to expanding A and R by refining and specializing. Formally, we can define these synthesis and analysis processes as follows.

Synthesis:

$\Xi = \langle S, V, C, R, A\rangle \Rightarrow \Xi' = \langle S \cup S', V \cup V', C \cup C', R, A\rangle$,

where S', V', and C' do not contradict the existing axioms of A under the given R; S' and C' are non-empty sets.

Analysis:

$\Xi = \langle S, V, C, R, A\rangle \Rightarrow \Xi' = \langle S, V, C, R \cup R', A \cup A'\rangle$,

where R' and/or A' are non-empty sets.

It is easy to see that specifying synthesis and analysis, we formally 'shape' the mechanisms of creative thought – divergent (synthesis) and convergent (analysis) thinking. Moreover, the introduced definition of the design process clearly explicates the dual nature of creativity that is so often emphasized by the cognitive scientists: either exploration or transformation alone does not lead to truly creative, novel solutions [1]. Creativity is the product of the complex evolution, where analysis is responsible for the exploration and evolution of the design model, while synthesis essentially effects the transformation of the language, with which the model is described.

It is important to note that design analysis and synthesis are defined on sign systems, which are theories but not models, and which are necessarily open. This allows for non-deterministic treatment of designs, where a sign system is seen as an evolving theory.

4. Towards computer-aided synthesis

The introduced model of the design process arranges the formalism for representing problem solving processes and interprets creativity at the domain level. To provide for understanding design creativity at the system level, this model should be put into a methodological context. Below, we propose a methodology of design synthesis.

It can be postulated that in modern manufacturing, designing new versions of existing products generally predominates over designing new products. As time to market continually shortens, design is increasingly becoming a re-design task. Under these circumstances, inspiration and creativity in design frequently acquire quite realizable and pragmatic meaning: inspiration is reduced to knowing what to adopt and borrow from previous experience, and creativity becomes in much a matter of how to reuse that which is borrowed.

In [7], Kryssanov *et al.* described a model of creative reasoning based on analog search, adoption, and conceptual blending. The model particularizes a cognitive mechanism of real-time similarity-based creativity. More precisely, the authors focused on the 'juxtaposition of dissimilar' cognitive mechanism – one of the two major mechanisms of similarity-based creativity (another one is known as 'deconceptualization') [2]. The underlying idea there was in moving away from normal associations during designing, putting and juxtaposing dissimilar concepts, objects, and structures in order to reconsider and recombine the objects to find a candidate solution. Based on the results of [7], we have developed a methodological pattern for computational modeling of the synthesis process in conceptual design as it is shown in Figure 2.

The synthesis process begins from formulation of design objectives and construction of the problem domain (i.e. the target) through identification of solutions satisfying the design functional requirements. An appropriate source domain (i.e. a domain, which is assumed to structurally be close to the target domain) is selected to serve as a source of potentially useful analogues. Two design sub-spaces are then constructed: one – to gather design solutions structurally compatible with the design under consideration (design analogues or metaphors), while another – to evoke and accumulate existing or newly generated solutions from the target domain. The source analogue concepts are matched with those of the target domain, and the analogue structure is decomposed if no target counterpart is available. The analogs found are then blended so that the source concepts provide a 'mould', and the target concepts – a 'priming' for the blend. It can be shown that in a general case, the synthesized design solution should be considered emergent inasmuch as it exhibits characteristics not perceivable in either of the input domains [7]. This solution needs to carefully be analyzed in respect to the initial specification.

It is understood that each of the information spaces depicted in the figure – the target domain, the source domain, the two parts of the design space, and the

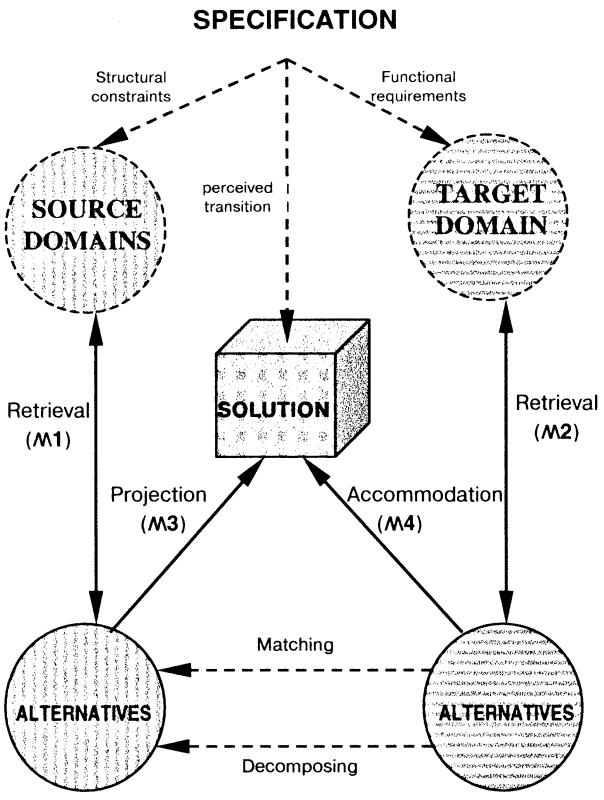


Figure 2. The computational model of the synthesis.

solution space – requires dealing with a distinct sign system. The model of the design process introduced in the previous section permits us to specify some of the important formal properties of the morphisms between these sign systems.

According to [6], morphism $\mathcal{M}$: $\Xi \Rightarrow \Xi'$ can be classified as

1) level preserving iff whenever sort s is lower level than sort s_1 in Ξ, $\mathcal{M}(s)$ has lower or equal level than $\mathcal{M}(s_1)$ in Ξ';

2) priority preserving iff whenever constructor c has higher priority than constructor c_1 in Ξ, $\mathcal{M}(c)$ has higher priority than $\mathcal{M}(c_1)$ in Ξ';

3) axiom preserving iff for each axiom a of Ξ, $\mathcal{M}(a)$ is a corollary of the axioms in Ξ'.

The development of the blend (a synthesized solution) can be controlled by adjusting the properties of the morphism compositions $\mathcal{M}3\circ\mathcal{M}1$ and $\mathcal{M}4\circ\mathcal{M}2$. It is obvious that $\mathcal{M}1$ and $\mathcal{M}2$ should be isomorphisms to ensure design backtracking, that is unavoidable during designing, and to provide for design information reuse. Through the analysis of a number of actual cases of conceptual designs in the field of electronics, it has been found in our research that a principal characteristic of $\mathcal{M}3$ is level and priority preserving, while $\mathcal{M}4$ is axiom preserving. It is also important to note that the analog matching between the concepts of the target and the source is to be done in a *bottom-up* manner. Besides, the analogue decomposition process necessarily has a *top-down* character.

6. Concluding remarks

In the presented work, we examined design creativity as the search for novel, potentially useful analogues with which to reinterpret the available 'traditional' solutions of problems similar to the one under consideration. In attempt to add more formality to the cognitive classical pattern of creativity, we suggested a model of the design creative process based on the synthesis-analysis interaction. Focusing on the particular cognitive mechanism of real-time similarity-based creativity, we proposed a computational model of the synthesis process and described some of its important properties. Therefore, this paper contributes to a better, mathematically supported understanding of the mechanisms of human creativity in design.

We should mention one caveat issue: we have not considered here the whole synthesis-analysis cycle, as the process of analysis has not been explicated at the system level. The crucial role of analysis, being opposite to the synthesis role, is to constrain the design conceptual space that makes creativity possible at all [1]. Currently, computer-based design tools and methodologies are overloaded by analysis, and in this sense, the issue of the development of a computational model of design analysis seems rather straightforward. The problem however is to integrate the synthesis and analysis models to formally define the creative process at the system level. We realize it a challenge for the future research.

We conclude this paper with a brief note on implementation. The described schemes of the creative process gave us a good insight for the development of an integrated support environment to assist human designers during conceptual design as it was presented in [7].

Acknowledgement

This research has been made within the 'Methodology of Emergent Synthesis' research project (No 96P00702) in the Program 'Research for the future' of the Japan Society for the Promotion of Science.

References

[1] Boden, M. (1995), Creativity and Unpredictability. *Stanford Humanities Review*, 'Constructions of the Mind' Special Issue edited by Stefano Franchi and Güven Güzeldere, Vol. 4, issue 2.

[2] Indurkhya, B. (1997), Computational Modelling of Mechanisms of Creativity, In: T. Veale (ed.) *Proceedings of the International Mind II Conference 'Computational Models of Creative Cognition'*, Dublin City University, September 15-16, John Benjamins Publisher, the Netherlands.

[3] O'Donoghue, D. (1997), An Integrated Model of Analogy for Creative Reasoning, In: T. Veale (ed.) *Proceedings of the International Mind II Conference 'Computational Models of Creative Cognition'*, Dublin City University, September 15-16, John Benjamins Publisher, the Netherlands.

[4] Herrmann, N. (1989), *The creative brain*, Brain Books (Rev. edition), ISBN: 0944850022.

[5] Goguen, J. and Malcolm, G. (1996), *Algebraic Semantics of Imperative Programs*. MIT, US.

[6] Goguen, J. (1999, to appear), An Introduction to Algebraic Semiotics, with Applications to User Interface Design. In: C. Nehaniv (ed.) *Computation for Metaphor, Analogy and Agents*, Springer Lecture Notes in Artificial Intelligence.

[7] Kryssanov, V.V., Tamaki, H., Ueda, K., and Svinin, M.M. (1998), Modelling Emergence to Support Engineering Design: A Cognitive Approach, In: *Proceedings of Rensselaer's International Conference on Agile, Intelligent, and Computer-Integrated Manufacturing*, CD paper, Rensselaer Polytechnic Institute, October 7-9, Troy, NY, USA.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Self-Organization Process in Interactive Manufacturing Environment

N. Fujii†, I. Hatono‡, K. Ueda††

†Graduate School of Science and Technology,
‡Information Processing Center,
††Department of Mechanical Engineering,
Kobe University, 1-1, Rokkodai, Nada-ku, Kobe 657-8501, Japan

Abstract

In this paper, we propose Interactive Manufacturing Systems, a new concept of manufacturing system based on Biological Manufacturing Systems. This proposal structure can deal with the difficulties arising from the increasing complexity and diversity in recent manufacturing environment. In interactive manufacturing systems, all participants in the process, e.g., a designer, a manufacturer, a consumer and a product, can influence each other during the whole life cycle of the products. We propose to use the self-organization principle and the virtual space so as to realize an interactive manufacturing system. Preliminary results of the self-organizing process in the three dimensional virtual space are presented, too.

Key Words: Self-Organization, Interactive Manufacturing, Virtual Space, Biological Manufacturing Systems

1 Introduction

Recently the complexity and dynamics in the societies, e.g., diversity of the culture, the individualization of lifestyle, assessment of the nature and so on, are increasing (Peklenik [1]). These factors cause qualitative and quantitative fluctuation in the internal and external manufacturing environment. Since the traditional manufacturing systems could poorly adapt to such the dynamic fluctuation, we have to explore further possibilities to create novel concepts of manufacturing. These days, next generation manufacturing systems such as Fractal or Holonic Manufacturing Systems are developed in order to overcome such difficulties in manufacturing.

Biological Manufacturing System (BMS) (Ueda et al. [4]) is also one of the new concepts in the manufacturing systems, which aims to deal with such complexities. BMS is based on biologically inspired ideas such as self-recognition, self-recovery, self-organization and evolution. In this study, the main objective is to propose and develop Interactive Manufacturing Systems based on the BMS (Ueda et al. [2]), in which all the participants can improve themselves through their interaction during the whole life cycle of the artifact, so as to conquer the complexities in manufacturing.

2 Interactive Manufacturing

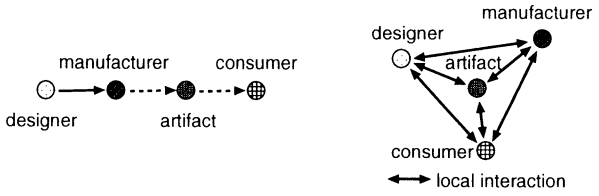
2.1 Difficulties in manufacturing

Generally speaking, an artifactual system is synthesized to satisfy a specification within a certain environment. These days, we face the difficulties in the process of the synthesis because the uncertainties are increasing in the environment or the specification. In the project, "Methodology of Emergent Synthesis", the difficulties are classified according to the uncertainties of the environment and the specification, and then various emergent methodologies are offered to overcome the difficulties (Ueda [3]).

In the manufacturing process, a product is also designed so as to fulfill its specification under a certain environment. Nowadays, the manufacturing environment is changing dynamically, for instance with the dynamic change of production demand. These facts lead not only to uncertainties in the manufacturing environment but also to difficulties to decide the initial specifications of the system, where we have to determine the specification of the system dynamically during the system is working. Hence, we should develop the novel manufacturing environment which makes possible that the participants in the process could interactively change the specification of the system.

2.2 Interactive manufacturing systems

In conventional manufacturing, all the processes are done in a linear manner, then the system could poorly adapt to the environmental fluctuations. We propose Interactive Manufacturing Systems based on BMS, in which all the participants, a designer, a manufacturer, a consumer and an artifact are interactively improved through the whole life cycle of the artifact (Fig. 1). Then the system would be able to adapt by being changed dynamically the specification in the case of fluctuation.



(a) Conventional linear Manufacturing (b) Interactive Manufacturing

Fig. 1: Interactive Manufacturing : from linear to interactive

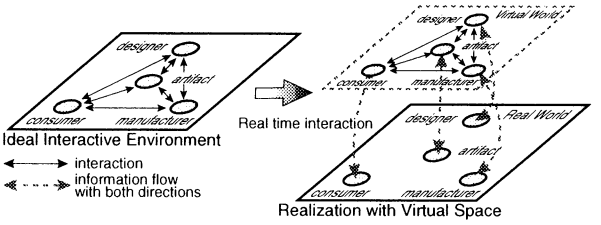


Fig. 2: Implementation of the *virtual* interactive space

2.3 Realization of interactive manufacturing

In realizing such interactive manufacturing environment, the following problems arise:

1. How can we realize the interaction among the entities in the environment?
2. How does the artifact react to the interaction by the human participants?

In this paper, we propose to use virtual space as an implementation. We, however, do not use the concept of virtual space in the only meaning of the mapping for the real space. We use virtual space as a mean where we could realize those matters that are impossible in the real space. For instance, the main feature of BMS, the DNA typed genetic information or the BN (Brain and Neuron) typed acquired information which make living creatures adaptive to the environmental factors, could be realized relatively easily.

In addition, we use the virtual space to control a real space by linking them with two directional interaction (Fig. 2). That is, first the real space is mapped to the virtual space, then the interaction among the participants occurs in the virtual space, and the result of the interaction influences the real space. Then the real space can not perform effectively without the concept of the virtual space.

We also propose to involve self-organization in order to allow the artifactual system be influenced by the human participants. We use here self-organization in the sense that global behavior emerges as the result of the local interactions between autonomous entities of which the system consists. When the environmental fluctuation or the interaction occur, re-organized behavior emerges, and as a result the system becomes adaptive.

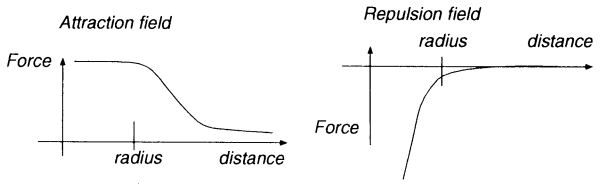


Fig. 3: Shapes of potential fields : divided into attraction and repulsion fields

Production process could be reactively changed if the interactive manufacturing environment is realized. The role of the conventional designer or manufacturer who could now only be involved with the design or production process will change, that is, they will be involved positively into the whole manufacturing process from design to disposal of the artifact.

3 Interactive Self-Organized Manufacturing System

Here, we focus especially on the production floor, as a part of interactive manufacturing, i.e., the relation of the manufacturer and the artifact. First, the implementation method of self-organization in the production floor will be described.

3.1 Self-organization in production process

We tried to apply self-organization to the production process (Vaario et al. [5]). In production floor, the autonomous entities are like machines and transporters. The production process is assumed to be the process that distributes materials among machines; the production process would be assumed as the local matching process between machines and transporters loaded with materials. Therefore, we consider self-organization in manufacturing where the global behavior, the material flow, emerges as the result of the multiple local interactions between machines and transporters.

How can the local interaction be implemented? We propose to use potential fields divided into attraction and repulsion fields, in which the physical forces represent the function of the distance between entities (Fig. 3). A machine generates attraction fields according to its capability, which kind of process it can do, on the other hand a transporter can sense the attraction fields according to the requirements of a material, which kind of process it wants to be done. Repulsion fields are also used to avoid collision among the entities.

There are several advantages of realizing the local interaction between entities with potential fields. First, they enable us to handle both of the time and space control in parallel. The time control means that the differences of priorities among the fields by

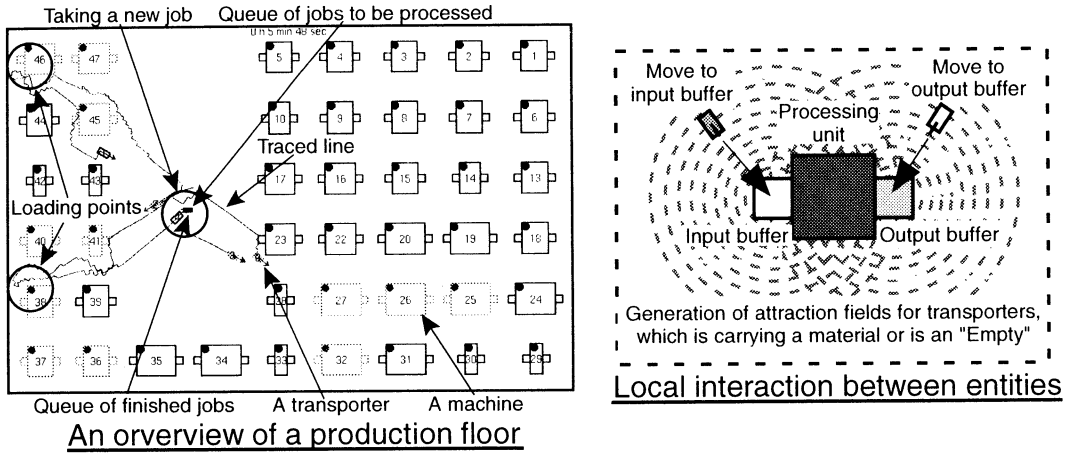


Fig. 4: Self-organization process with potential fields

discriminating the strength lead to more appropriate production schedule. On the other hand, space control means path planning of the material flow after deciding which machine is allocated to a material. Moreover, although the whole information of the system and the inputs of it should be well known to the common production scheduling process, that kind of information is not needed in the potential field method because the whole process is laid down as resulting from the multiplication of local interactions. This also provides adaptability of the system.

Fig. 4 illustrates the self-organization process in the production floor. There are 44 machines and 5 transporters in the floor, and the autonomous behavior of the transporters could be observed as resulting from the local interactions. Adaptive behavior could be also observed when internal environmental fluctuations such as the malfunction of some machines occur.

3.2 Implementation

The method of self-organization is connected with virtual reality to implement the interactive self-organized manufacturing system (Vaario et al. [6]). Fig. 5 represents the overview of the implementation. The system is not a single big one but distributed system with two modules, virtual reality (VR) and self-organization simulation (SOS). SOS is responsible for the factory operation, on the other hand, VR is responsible for three dimensional visualization and human interaction. They are connected via the communication software, Message Passing Interface (MPI), and communicate each other exchanging messages, e.g., the position and the state of the machines and transporters, the position of the human participant and so on.

There are some advantages of creating the system with two distributed modules: 1) Each module could be developed individually. 2) They could be executed in the different computer systems. 3) The computational load could be decreased.

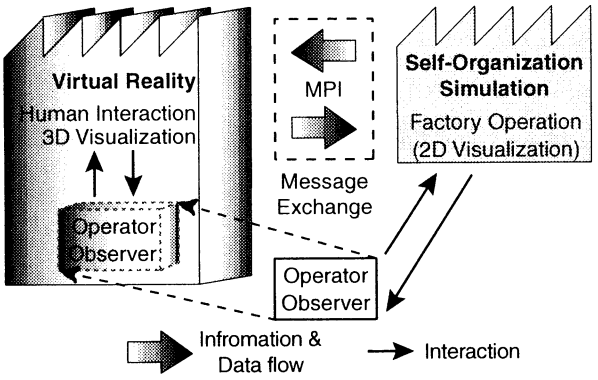


Fig. 5: Implementation overview of the integration of self-organization simulation with virtual reality

3.3 Preliminary results

Preliminary results available are shown in Fig. 6. The whole simulation process runs in three dimensional virtual space. The system involves abilities for human interaction; a human participant can freely walk through the production floor and observe the production process directly.

An example of human interaction is also available in Fig. 6. Both figures indicate the viewpoint of the participant in the production floor. In this case, he is a production engineer, who can change the floor layout, the position of the machines, in real time during the system is running. While the he left part in Fig. 6 shows the initial configuration of the machines, the production engineer has changed the position of a machine on the right figure. That is, the production engineer can influence the production process in real time.

The system will react correctly to the action of the production engineer. In Fig. 7, a transporter reaches

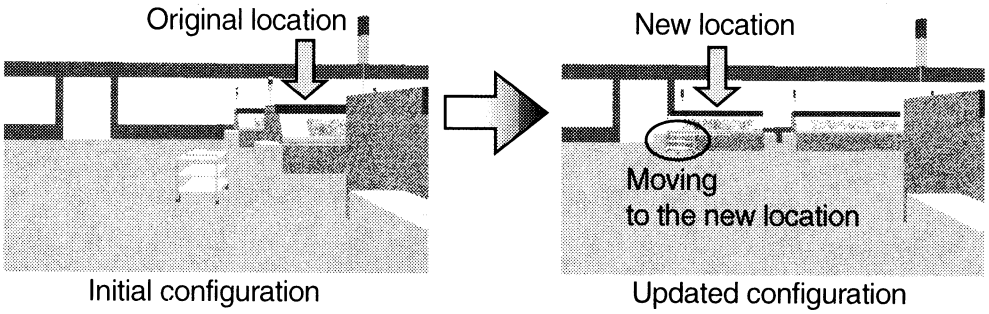


Fig. 6: An example of changing the location of a machine by a production engineer acting in the virtual space

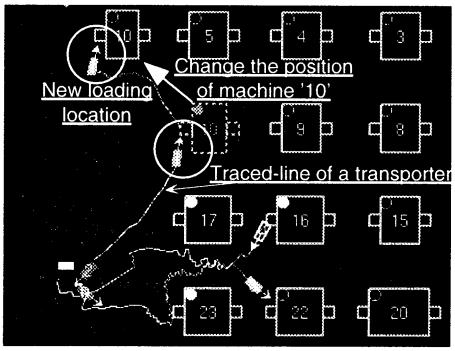


Fig. 7: An example of reactive behavior of the system: the action by the production engineer has been realized

the new position of the machine and delivers its load to the correct place as the result of self-organization.

3.4 Discussion

The realized interactions are quite simple but interesting with respect to the relations among the designer, the production engineer, the consumer and the artifact. When the demands or the design of the artifact are changed by the consumer or the designer, in the interactive environment the production engineer does not need to stop the production line in order to change the configuration of the line. He can change the configuration of the floor layout directly and quickly in the face of various fluctuations; the system incorporates flexibility towards dynamic environmental changes.

4 Summary

We have introduced a novel manufacturing system, interactive manufacturing, and presented an implementation with self-organization principle. Preliminary results were also presented, where simple interactions between the production engineer and the system could be realized.

Acknowledgement

This study was supported in part by the project of "Methodology for Emergent Synthesis" in Research for the Future Program of the Japan Society for the Promotion of Science (JSPS), JSPS Research Fellowships for Young Scientists and IMS/NGMS Project. Also a part of this study includes collaboration with Dr. Vaario (Nokia Research Center, Tokyo).

References

[1] Peklenik, J. (1997), "Structural and Operational Complexity of Future Manufacturing Systems", New Manufacturing Era, 29th CIRP International Seminar on Manufacturing Systems, Osaka, Japan, pp. 1-8

[2] Ueda, K., Vaario, J. and Fujii, N. (1998), "Interactive Manufacturing : Human Aspects for Biological Manufacturing Systems", Annals of the CIRP, 47(1), pp. 389-392

[3] Ueda, K. (1998), "Introduction of the Project 'Methodology of Emergent Synthesis' ", Proc. of Workshop on the Methodology of Emergent Synthesis(WMES'98), Kobe, Japan, pp. 1-4

[4] Ueda, K. and Ohkura, K. (1995), "A Biological Approach to Complexity in Manufacturing Systems", Proc. of 27th CIRP International Seminar on Manufacturing, Ann Arbor, pp. 69-78

[5] Vaario, J. and Ueda, K. (1997), "An Emergent Modeling Method for Dynamic Scheduling,", Proc. of the Second World Congress on Intelligent Manufacturing Processes & Systems, Budapest, Hungary, pp. 187-198

[6] Vaario, J., Fujii, N., Scheffter, D. et al. (1997), "Factory Animation by Self-Organization Principles", Proc. of International Conference on Virtual Systems and Multi Media(VSMM'97), Geneva, Switzerland, pp. 235-242

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Intelligent Control of Robot Manipulator by Visual Feedback

Sung-Hyun Han[1], Jong-il Bae[2] and Man-Hyung Lee[3]

1. School of Mechanical Engineering, Kyungnam Univ., Masan City, Korea.
   E-mail: shhan@kyungnam.ac.kr, TEL: +82-551-49-2624, FAX: +82-551-43-8133
2. Dept. of Electrical Engineering, Pukyung National Univ., Pusan City, Korea.
   E-mail: jibae@pine.pknu.ac.kr, TEL: +82-51-620-1437, FAX: +82-51-620-1437
3. School of Mechanical Engineering, Pusan National Univ., Pusan City, Korea.
   E-mail: mahlee@hyowon.cc.pusan.ac.kr, TEL: +82-51-510-1456, FAX: +82-51-510-2331

Abstract

This paper presents a new approach to visual servoing with the stereo vision. In order to control the position and orientation of a robot with respect to an object, a new technique is proposed using a binocular stereo vision. The stereo vision enables us to calculate an exact image Jacobian not only at around a desired location but also at the other locations. The suggested technique can guide a robot manipulator to the desired location without giving such priori knowledge as the relative distance to the desired location or the model of an object even if the initial positioning error is large. This paper describes a model of stereo vision and how to generate feedback commands. The performance of the proposed visual servoing system is illustrated by the simulation and experimental results and compared with the case of conventional method for a SCARA robot.

KEYWORDS : Real-time control, SCARA robot, Visual servoing, Feedback command, Image Jacobian

1. Introduction

There are mainly two ways to put the visual feedback into practice. One is called look-and-move and the other is visual servoing. The former is the method which transforms the position and orientation of an object obtained by a visual sensor into those in the world frame fixed to an environment and guides the arm of the manipulator to a desired location in the world frame.[1,2] In this method, precise calibration of a manipulator and camera system is needed. On the contrary, visual servoing uses the Jacobian matrix which relates the displacement of an image feature to the displacement of a camera motion and performs a closed-loop control regarding the feature as a scale of the state. Therefore, we can construct a servo system based only on the image and can have a robust control against the calibration error because there is no need to calculate the corresponding location in the world frame.[2,3,4] A hand eye system is often used in visual feedback and there are two ways of arranging the system. One is placing a camera and a manipulator separately; the other is placing the camera at the end-tip of the manipulator. The former motion strategy of the manipulator becomes more complicated than the latter. In the latter, it is easy to control the manipulator using a visual information because the camera is mounted on the manipulator end-tip. In this paper, we deal with the latter method. In the conventional works, some researches have presented methods to control the manipulator position with respect to the object or to track the feature points on an object using a hand eye system as the application of visual servoing.[3,4] These methods maintain or accomplish a desired relative position between the camera and the object by monitoring feature points on the object from the camera.[5,6]

However, these have been all done by the hand eye system with monocular visions and it is necessary to compensate for the loss of information because the original three-dimensional information of the scene is reduced to two-dimension information on the image. For instance, we must add an information of the three-dimension distance between the feature point and the camera in advance or use a model of object stored in the memory. Besides, a problem that the manipulator position fails to converge to a desired value arises depending on the way of selecting feature points or when the initial positioning error is not small. It is because some elements of the image Jacobian cannot be computed with only the information of the image and substituting approximate values at the desired location for them may result in large errors at the other locations.

This paper presents a method to solve this problem by using a binocular stereo vision. The use of stereo vision

can lead to an exact image Jacobian not only at around a desired location but also at the other locations. The suggested technique places a robot manipulator to the desired location without giving such priori knowledge as the relative distance to the desired location or the model of an object even if the initial positioning error is large. This paper deals with modeling of stereo vision and how to generate feedback commands. The performance of the proposed visual servoing system was evaluated by the simulations and experiments and obtained results were compared with the conventional case for a SCARA robot.

2. Stereo Vision Model

We define the frame of a hand-eye system with the stereo vision and use a standard model of the stereo camera whose optical axes are set parallel each other and perpendicular to the baseline. The focal points of two cameras are apart at distance d on the baseline and the origin of the camera frame $\sum_c$ is located at the center of these cameras. Fig. 1 represents the schematic diagram of a suggested visual servoing system. In Fig. 1 two DSP vision boards (MVB03) are used, which were made Samsung Electronics Company in Korea based-on the TMS320C30 chips.

An image plane is orthogonal to the optical axis and apart at distance f from the focal point of a camera and the origins of frame of the left and right images , $\sum_l$ and $\sum_r$, are located at the intersecting point of the two optical axes and the image planes. The origin of the world frame $\sum_w$ is located at a certain point in the world. The x, y, and z axes of the coordinate frames are shown in Fig. 2.

Now let ${}^l p = ({}^l x, {}^l y)$ and ${}^r p = ({}^r x, {}^r y)$ be the projections onto the left and right images of a point p in the environment, which is expressed as ${}^c p = ({}^c x\ {}^c y\ {}^c z)^T$ in the camera frame. Then the following equation is obtained (see Fig. 2).

$$ {}^l x\ {}^c z = f({}^c x + 0.5d) \quad (1\text{-a}) $$

$$ {}^r x\ {}^c z = f({}^c x - 0.5d) \quad (1\text{-b}) $$

$$ {}^l y\ {}^c z = f\ {}^c y \quad (1\text{-c}) $$

$$ {}^r y\ {}^c z = f\ {}^c y \quad (1\text{-d}) $$

Suppose that the stereo correspondence of feature points between the left and right images are found. In the visual servoing, we need to know the precise relation between the moving velocity of camera and the velocity of feature points in the image, because we generate a feedback command of the manipulator based on the velocity of feature points in the image.

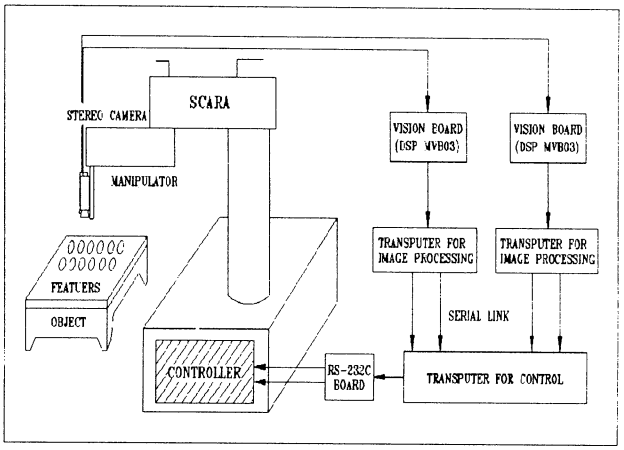


Fig. 1 Schematic diagram of visual servoing system

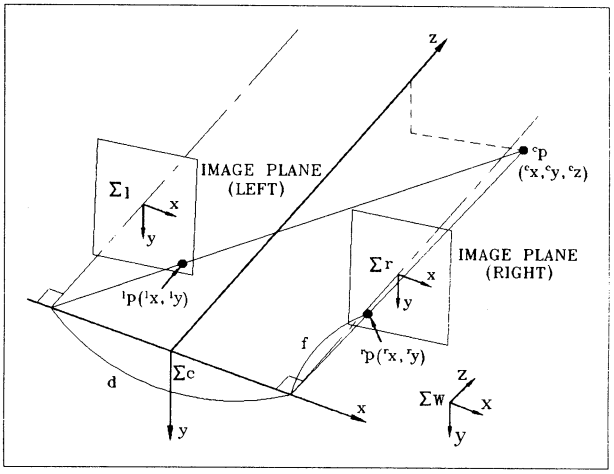


Fig. 2 The coordinates system of stereo vision model.

This relation can be expressed in a matrix form which is called the image Jacobian. Let us consider n feature points $p_k (k = 1, \cdots, n)$ on the object and the coordinates in the left and right images are ${}^l p_k({}^l x_k, {}^l y_k)$ and ${}^r p_k({}^r x_k, {}^r y_k)$, respectively. Also define the current location of the feature points in the image ${}^I p$ as

$$ {}^I p = ({}^l x_1\ {}^r x_1\ {}^l y_1\ {}^r y_1\ {}^r y_1 \cdots {}^l x_n\ {}^r x_n\ {}^l y_n\ {}^r y_n)^T \quad (2) $$

where each element is expressed with respect to the virtual image frame $\sum_P$.

First, to make it simple, let us consider a case when the number of the feature points is one. The relation between the velocity of feature point in image ${}^I \dot{p}$ and the velocity of camera frame ${}^c \dot{p}$ is given as

$$ {}^I \dot{p} = {}^P J_c\ {}^c \dot{p} \quad (3) $$

where ${}^P J_c$ is the Jacobian matrix which relates the two frames. Now let the translational velocity components of camera be σ_x, σ_y, and σ_z and the rotational velocity components be w_x, w_y, w_z, then we can

express the camera velocity V as

$$V = [\sigma_x \ \sigma_y \ \sigma_z \ w_x \ w_y \ w_z]^T = [{}^c v_c \ {}^c w_c]^T \quad (4)$$

Then the velocity of the feature point seen from the camera frame ${}^c\dot{p}$ can be written

$$\begin{aligned} {}^c\dot{p} &= \frac{d^c p}{dt} \\ &= \frac{d}{dt}\,{}^cR_w({}^w p - {}^w p_c) \\ &= {}^cR_w\{-{}^w w_c \times ({}^W p - {}^w p_c)\} + {}^cR_w({}^W\dot{p} - {}^w\dot{p}_c) \end{aligned} \quad (5)$$

where ${}^cR_w$ is the rotation matrix from the camera frame to the world frame and ${}^w p_c$ is the location of the origin of the camera frame written in the world frame. As the object is assumed to be fixed into the world frame, ${}^w\dot{p}=0$. The relation between ${}^c\dot{p}$ and V is

$$\begin{aligned} {}^c\dot{p} &= {}^cR_w\{-{}^w w_c \times ({}^w p - {}^w p_c)\} - {}^cR_w\,{}^w\dot{p}_c \\ &= -{}^c w_c \times {}^c p - {}^c\dot{p}_c \\ &= \begin{bmatrix} -w_y\,{}^c z + w_x\,{}^c y - \nu_z \\ -w_z\,{}^c x + w_x\,{}^c z - \nu_y \\ -w_x\,{}^c y + w_y\,{}^c x - \nu_z \end{bmatrix} \end{aligned} \quad (6)$$

Therefore, substituting Eq. (6) into Eq. (3), we have the following equation.

$$\begin{aligned} {}^I\dot{p} &= {}^I J_c \ {}^c\dot{p} \\ &= J\ V \end{aligned} \quad (7)$$

In Eq. (7) matrix J which expresses the relation between velocity ${}^I\dot{p}$ of the feature point in the image and moving velocity V of the camera is called the image jacobian.

From the model of the stereo vision Eq. (1), the following equation can be obtained.

$${}^c x\,(2\,{}^l x - {}^r x) = d\,{}^l x + {}^r x \quad (8)$$

$${}^c y\,({}^l x - {}^r x) = {}^l y\ d = {}^r y\ d \quad (9)$$

$${}^c z\,({}^l x - {}^r x) = f\,d \quad (10)$$

Above discussion is based on the case of one feature point. In practical situation, however, the visual servoing is realized by using plural feature points. When we use n feature points, image Jacobian $J_1, \cdots, J_n$ are given from the coordinates of feature points in the image. By combining them, we express the image Jacobian as

$$J_{im} = [J_1 \cdots\cdots J_n]^T \quad (11)$$

Then, it is possible to express the relation of the moving velocity of the camera and the velocity of the feature points even in the case of plural feature points, that is,

$${}^I\dot{p} = J_{im}\ V \quad (12)$$

where we suppose that the stereo and temporal correspondence of the feature points are found.

In the case of the monocular, the image Jacobian J has the following form.

$$J = f\begin{bmatrix} -\frac{1}{{}^c x} & 0 & \frac{{}^c x}{{}^c x^2} & \frac{{}^c x\,{}^c y}{{}^c z} & -(1+\frac{{}^c x^2}{{}^c x^2}) & \frac{{}^c y}{{}^c z} \\ 0 & -\frac{1}{{}^c x} & \frac{{}^c y}{{}^c x^2} & 1+\frac{{}^c y^2}{{}^c z^2} & -\frac{{}^c x\,{}^c y}{{}^c x^2} & -\frac{{}^c x}{{}^c z} \end{bmatrix} \quad (13)$$

We now introduce the positional vector of the feature point in the image of monocular vision using the symbol ${}^m P = ({}^m x,\ {}^m y)$. This is the projection of the point expressed as ${}^c P = ({}^c x\ {}^c y\ {}^c z)^T$ in the camera frame into the image frame of the monocular vision, and has the following relation.

$${}^m x = f\ {}^c x\ {}^c z^{-1} \quad (14\text{-a})$$

$${}^m x = f\ {}^c y\ {}^c z^{-1} \quad (14\text{-b})$$

Substituting Eq.s (14-a) and (14-b) into Eq. (13) yields another expression of the image Jacobian for the monocular vision.

$$J = f\begin{bmatrix} -\frac{1}{{}^c z} & 0 & \frac{{}^m x}{{}^c z} & \frac{{}^m x\,{}^m y}{f} & -\frac{{}^m x^2+f^2}{f^2} & {}^m y \\ 0 & -\frac{f}{{}^c x} & \frac{{}^m y}{{}^c x} & 1+\frac{{}^m y^2+f^2}{f^2} & -\frac{{}^m x\,{}^m y}{f} & -{}^m x \end{bmatrix} \quad (15)$$

A disparity which corresponds to the depth of the feature point, is included in J in the case of the stereo vision, but s-term expressed in the camera frame ${}^c z$ is included in J in the case of the monocular vision.

3. Feedback Command

In the visual servoing, the manipulator is controlled so that the feature points in the image reach their respective desired locations.

We define an error function between the current location of the feature points in image ${}^I p$ and the desired location ${}^I p_d$ as

$$E = Q({}^I p - {}^I p_d) \quad (16)$$

where Q is a matrix which stabilizes the system. Then the feedback law is defined as following equation

$$V = -G E \tag{17}$$

where G corresponds to a feedback gain.

To realize the visual servoing, we must choose Q so that convergence is satisfied with the error system can be satisfied with

$$\begin{aligned} \dot{E} &= \frac{\partial E}{\partial t} \\ &= Q\frac{\partial\, {}^{I}\dot{p}}{\partial t} \\ &= Q\, {}^{I}\dot{p} \\ &= Q J_{im} V \\ &= -G Q J_{im} E \end{aligned} \tag{18}$$

We use pseudo-inverse matrix of the image Jacobian J_{im} for Q to make $Q J_{im}$ positive and not to make an input extremely large, that is,

$$Q = J_{im}^{+} = (J_{im}^{T} J_{im})^{-1} J_{im}^{T} \tag{19}$$

Therefore, the feedback command is given as

$$V = -G J_{im}^{+} ({}^{I}p - {}^{I}p_d) \tag{20}$$

Fig. 3 shows a block diagram of the control scheme described by Eq. (20). Note that the feedback command u is sent to the robot controller and both the transformation of u to the desired velocity of each joint angle $\dot{q}_d$ and its velocity servo are accomplished in the robot controller as show in Fig. 3.

Futhermore, as J_{im} is a $4n \times 6$ matrix and pseudo-inverse matrix J_{im}^{+} is a $6 \times 4n$ matrix, a feedback command Eq. (20) of 6 degrees of freedom is obtained.

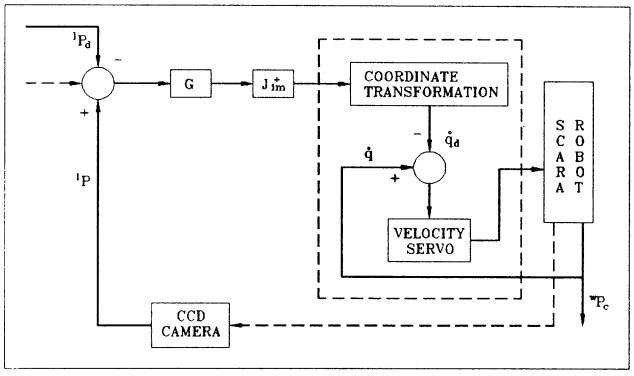


Fig. 3 Block diagram of visual feedback system.

4. Simulation and Experiments

4.1 Simulation

We have compared the visual servoing using the monocular vision with that using the stereo vision by the simulation. In the simulation, feature points of an object are the four corners of a square whose side dimension is 300 mm. In the same condition, we used four feature points even in the stereo vision. Parameters used the focal length, f=16 mm, baseline d=130 mm, sampling time of 50 msec, gain λ=1, desired location ${}^{c}P_d = (100\ 100\ 500)^{T}$mm desired orientation in Euler angle $(\varphi, \theta, \psi) = (0,0,0)$ rad, initial error $(-50\ -50\ -50)^{T}$ mm in the translation and $(\varphi, \theta, \psi) = (20,20,20)$ rad in the orientation.

error image. We select the four corners of a rectangle whose size is 200×200 mm as the feature points and set the translational error as (-150 -150 -450) mm and the other values are the same as before (Fig. 4).

In Fig. 4, we can see that the result diverges in the case of the monocular vision, but converges in the case of the stereo vision. This is because the image Jacobian is fixed at the desired location in case of the monocular vision. Therefore, a correct feedback command can not be generated when the initial error is large. On the other hand, the image Jacobian can be updated at every in the case of the stereo vision, thus it is possible to generate a correct feedback command which assures the stability visual servoing.

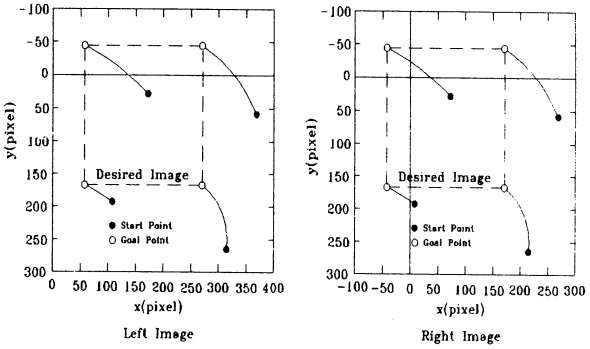


Fig. 4 Trajectories of the feature points on the images.

4.2 Experiments

In experiments, we used a four-axis SCARA Robot (SM5 Model) made in Korea with a stereo camera attached to the end tip of the arm. The feature points are three circular planes of 20 mm radius on three corners of a equilateral triangle, one side 87 mm and are placed on the board. Precise calibration had not been done for the stereo camera attached to the end-tips.

Two stereo images were taken and transformed to the binary images in the real time and in parallel by two image input devices and the coordinate of the gravitational center of each feature point was calculated in parallel by two transputers. We gave the stereo correspondence of the feature point in the first sampling.

However, the stereo and temporal correspondence of the feature points in the succeeding sampling were found automatically by searching a nearby area where there were the feature points in the previous sampling frame. The coordinates of the feature points were sent to a transputer for motion control and it calculated a feedback command for the robot. The result was sent to the robot controller by using RS-232C, and the robot was controlled by a velocity servo system in the controller.

The sampling period of visual servoing was about 50 msec. Details were 16 msec for taking a stereo images, about 1 msec for calculating the coordinates of the feature points, 3 msec for calculating feedback command, about 16 msec for communicating with the robot controller. If we send a feedback input to the robot controller without using RS-232C, the faster visual servoing can be realized.

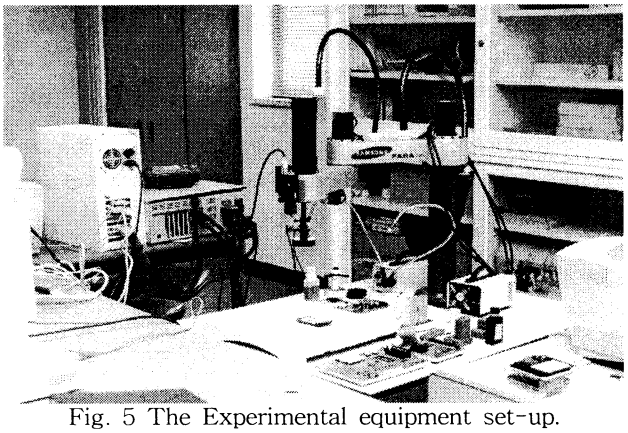
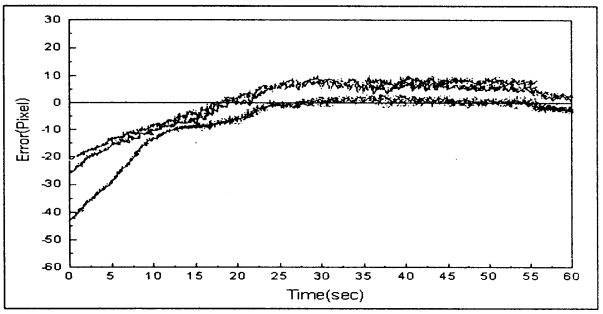


Fig. 5 The Experimental equipment set-up.

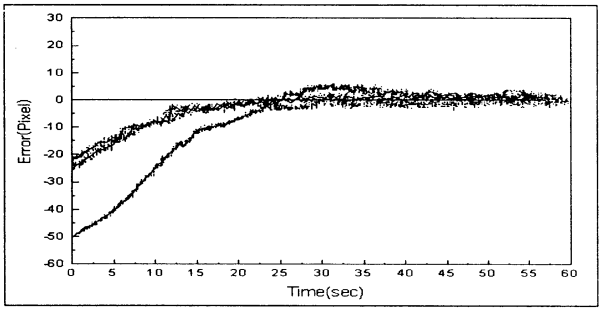
The desired location was $(0, 0, 500)^T$ mm and the desired orientation in Euler angle, $(\phi, \theta, \psi) = (0, 0, 0)$ degree and the initial error was $(50, 50, 50)^T$ mm for translation. The other parameters were the same as in the simulation. The error of current and desired location of the feature points are shown in Fig. 7 From these experimental results, we can see that the manipulator converges toward a desired location even if the calibration is not precise.

5. Conclusion

This paper proposes a new method of visual servoing with the stereo vision to control the position and orientation of a SCARA robot with respect to an object. The method overcomes the several problems associated with the visual servoing with the monocular vision. By using the stereo vision, the image Jacobian can be calculated at any position. So neither shape information nor desired distance of the target object is required. Also the stability of visual servoing is assured even when the initial error is very large. We have shown the effectiveness of this method by simulation and experiments.



(a) Left image



(b) Right image

Fig. 6 Position Error in x and y axes.

To use this visual servoing in practical tasks, there still exist many problems such as the number of feature points to reduce noise or the quantization error and the way to choose feature points. Nevertheless, this method overcomes the several problems in visual servoing with the monocular vision.

References

1. Allen, P. K., Toshimi, B., Timcenko, A., "Real-Time Visual Servoing". In Proceeding of the IEEE International Conference on Robotics and Automation(1991), 851-856.
2. Hashimoto, K., Kimoto, T. Edbine, T., and Kimura, H., "Manipulator control with image-based visual servo". In Proceedings of IEEE International Conference on Robotics and Automation(1991), 2267-2272.
3. Chaumette, F., Rives, P., and Espiau, B., "Positioning of a robot with respoect to an object, tracking it and estimating its velocity by visual servoing". In Proceedings of the IEEE International Conference on Robotics and Automation(1991), 2248-2253.
4. Hashimoto, K., Edbine, T., and Kimura, H., "Dynamic visual feedback control for a hand-eye manipulator". In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems(1992), 1863-1868.
5. Bernard, E., Francois, C., and Patrick, R., "A new approach to visual servoing in robotics". IEEE Transactions on Robotics and Automation(1992), 313-326.
6. Weiss, L. E., Sanderson, A. C., and Neuman, C. P., "Dynamic sensor-based control of robots with visual feedback". IEEE Journal of Robotics and Automation(1987), 404-417.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Robust Predictive Control of Robot Manipulators

Myung-Chul Han, Man Hyung Lee

School of Mechanical Engineering, Pusan National University
San 30, Jangjeon-Dong, Kumjung-Ku, Pusan 609-735, Korea
e-mail : mchan@hyowon.cc.pusan.ac.kr
mahlee@hyowon.cc.pusan.ac.kr

Abstract

In this paper, we present a predictive control approach to the robust robot control whose use is made of the Lyapunov min-max approach. Since the control design of a real manipulator system may often be made on the basis of the imperfect knowledge about the model, it is an important trend to design a robust control law that will guarantee the desired performance of the manipulator under uncertain elements. In the preceeding robust control work, we choose several control parameters in the admissible set where the desired stability can be achieved. As we introduce a optimal predictive control technique in robust control here, we can consider both the stability and the performance of manipulators. A new class of robust controls which are combined with a optimal predictive control are constructed.

1. Introduction

In the control design of real robot manipulators, one of main obstacles is the difference between a mathematical model and the real system. The difference may arise from e.g., improper choice of system parameters, friction, varying payload, disturbance, etc. The robust control approach is to solve this problem as uncertainties are introduced in a model and the controller can achieve the desired properties in spite of the imperfect modeling. The deterministic robust control design of manipulators can be found in, e.g., Chen[1], Chen and Pandey [2], Reithmeier and Leitman[3], Shoureshi et al.[4], Han[5,6], and their bibliographies.

The concept of predictive control was introduced in the late 70s and has received great attention since then. Predictive control strategy is to forecast the process output and to set up the several control scenarios that drive the future plant responses to track the reference inputs, and to select the best one. Predictive control has shown good control results for the system with variable parameters, variable dead time, and model order change[7,8,9]. Most of predictive control schemes have been developed for the linear system. Lu[10] has extended the concept to nonlinear control system and evaluated the stability and robustness. The predictive control concept of nonlinear systems is to similar to that of linear systems, i.e., control system predicts the future response of the system based on the prediction model obtained from appropriate functional expansions and then, computes the control law minimizing the local difference between the predicted and desired responses.

In this work, we propose a new class of robust predictive control of manipulator where an optimal predictive control is merged in robust control. In the robust predictive control, optimal predictive scheme is adopted to evaluate the system performance and stability and robust control scheme is to improve the stability and robustness. The stability analysis for the manipulator with the proposed controller is also addressed.

2. Robot dynamics and stability

The dynamic equation of an n-link rigid manipulator can be written as

$$M(q)\ddot{q}+C(q,\dot{q})\dot{q}+g(q)+f(q,\dot{q},t)=\tau \quad (1)$$

where q, $\dot{q}$, $\ddot{q}$ are the n-dimensional vectors of joint positions, velocities, and accelerations, respectively, and τ is n-dimensional vector of joint torques supplied by the actuator. Furthermore, $M(q)$ is the $n\times n$ inertia matrix, $C(q,\dot{q})\dot{q}$ is the Coriolis and centrifugal terms, $g(q)$ is the gravitational term, and $f(q,\dot{q},t)$ is the torque arising due to the structured and unstructured uncertain terms, that may include friction forces and external disturbances.

Remark 1. The inertia matrix $M(q)$ is uniformly positive definite, symmetric, and bounded. That is, there exist constants m_1, $m_2 > 0$ such that $m_1 I \le M(q) \le m_2 I$, $\forall q \in R^n$. The nonlinear term $C(q,\dot{q})$ in (1) can be suitably chosen such that $\dot{M}(q)-2C(q,\dot{q})$ is skew symmetric in the

sense that for any $\zeta \in R^n$, $\zeta^T(\dot{M}(q)-2C(q,\dot{q}))\zeta=0$, $\forall q, \dot{q} \in R^n$.

In this work, the following definition describes the desired system behavior.

Definition 1.[1,2,5,6,12,13,14] Consider a nonlinear system $\dot{x}(t)=a(x(t),\sigma(t),t)$ where $\sigma(t)$ is a vector of uncertainties. We say that the system is practically stable if the following properties are satisfied.

property 1. Existence of solutions: The system possesses a solution .

property 2. Uniform boundedness: Given any $\underline{r} \in (0,\infty)$ and any solution $x(\cdot)$: $[t_0,\infty) \rightarrow R^n$, there exists a $d(\underline{r})<\infty$ such that $\|x_0\| \leq \underline{r}$ implies $\|x(t)\| \leq d(\underline{r})$ for all $t \in [t_0,\infty)$.

property 3. Uniform ultimate boundedness: Given any $\bar{r} \geq r_0$ and any $\underline{r} \in (0,\infty)$, there exists a finite time $T(\underline{r},\bar{r})$ such that $\|x_0\| \leq \underline{r}$ implies $\|x(t)\| \leq \bar{r}$ for all $t \geq t_0+T(\underline{r},\bar{r})$.

property 4. Uniform stability: Given any $\bar{r} \geq r_0$, there exists a $\delta(\bar{r})>0$ such that $\|x_0\| \leq \delta(\bar{r})$ implies $\|x(t)\| \leq \bar{r}$ for all $t \geq t_0$.

3. Robust predictive control

The predictive control approach was introduced firstly in the late 70s and has received great attention since then. Predictive control strategy is to forecast the process output and to set up the several control scenarios that drive the future plant responses to track the reference inputs, and to select the best one. The predictive control concept of nonlinear systems is to similar to that of linear systems, i.e., control system predicts the future response of the system based on the prediction model obtained from appropriate functional expansions and then, computes the control law minimizing the local difference between the predicted and desired responses.

We first consider an optimal predictive control of the following type systems whose category the motion equation of manipulator can belong to:

$$\ddot{x}_1 = f(x_1, \dot{x}_1, t) + B(x_1, \dot{x}_1, t)u \tag{2}$$

where $x_1 \in R^n$ and $u \in R^n$ is the control. It can be rewritten in state space:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x,t) + g(x,t)u \end{aligned} \tag{3}$$

where $x_2 \equiv \dot{x}_1$ and $x \equiv (x_1^T\ x_2^T)^T \in R^{2n}$ is the state vector.

For a small h, a future output $x_2(t+h)$ can be expanded by Taylor series(to the order of the system) as

$$x_2(t+h) \cong x_2(t) + h\dot{x}_2(t), \tag{4}$$

The first order differentiation term in right hand side of Eq.(4) is replaced by the dynamic equation in Eq. (2) and the resulting expression is as follows

$$x_2(t+h) \cong x_2(t) + h(f+Bu) \tag{5}$$

Similarly

$$x_1(t+h) \cong x_1(t) + hx_2(t) \tag{6}$$

At this point, we assume that the desired trajectory $x_d(t+h)$ can be predicted by the following formula with small error

$$x_d(t+h) \cong x_d(t) + h\ \dot{x}_d(t) \tag{7}$$

In order to find the current control u that improves tracking accuracy at the next instant, we consider a pointwise minimization performance index that penalizes the tracking error at $t+h$ and current control expenditure

$$\begin{aligned} J(u) = \ & \frac{1}{2}[x(t+h)-x_d(t+h)]^T Q \\ & [x(t+h)-x_d(t+h)] + \frac{1}{2}u^T R u \end{aligned} \tag{8}$$

where $Q = \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix}$, $Q_1 \in R^{n \times n}$ and $Q_2 \in R^{n \times n}$ are positive, and $R \in R^{n \times n}$ is positive semidefinite. Replace $x(t+h)$ and $x_d(t+h)$ in (8) by predictions (5)-(6), respectively. For given $x(t)$ and $x_d(t)$, the unique control that minimizes J is obtained from $\partial J/\partial u=0$ is as follows[10]

$$\begin{aligned} u = \ & -B^{-1}PB^{-T}\left\{\frac{1}{2h^2}B^T Q_1[e_1 + h\dot{e}_1 \right. \\ & \left. + \frac{h^2}{2}(f - \ddot{x}_{2d})] + \frac{1}{h}B^T Q_2[e_2 + h(f - \dot{x}_{2d})]\right\} \end{aligned} \tag{9}$$

where $e = x - x_d = [e_1^T\ e_2^T]^T$ and $P \equiv (1/4Q_1 + Q_2)^{-1} > 0$.

Now we apply the result (9) to the robot manipulator. The manipulator equation (1) can be rewritten as follows

$$\ddot{q} = -M^{-1}(C\dot{q} + g + f) + M^{-1}\tau \tag{10}$$

In the format (2), q, $-M^{-1}(C\dot{q}+g+f)$, and M^{-1} in (10) correspond to x_1, f, and B in (2).

Substituting (9) into (3) yields the stable error dynamics

$$\dot{e}=\begin{bmatrix} 0 & \\ -\frac{1}{2h^2}(1/4Q_1+Q_2)^{-1}Q_1 & \\ I & \\ -\frac{1}{h}(1/4Q_1+Q_2)^{-1}(1/2Q_1+Q_2) & \end{bmatrix} e \equiv \Lambda e \quad (11)$$

When the plant parameter functions vary or the uncertainty exist in the dynamics, we do not know the f and B, but have the estimated functions $\hat{f}$ and $\hat{B}$. Then the control input u based on $\hat{f}$ and $\hat{B}$ has an additional term as follows

$$u=-\hat{B}^{-1}P\hat{B}^{-T}\left\{\frac{1}{2h^2}\hat{B}^T Q_1\left[e_1+h\dot{e}_1+\frac{h^2}{2}(\hat{f}-\ddot{x}_{2d})\right]+\frac{1}{h}\hat{B}^T Q_2[e_2+h(\hat{f}-\ddot{x}_{2d})]\right\}+\hat{B}^{-1}u_r \quad (12)$$

where u_r is the robust control term that will be given later. Then the error dynamics are

$$\dot{e}=\Lambda e+\Gamma(u_r+\eta) \quad (13)$$

where

$$\Gamma=\begin{bmatrix}0\\ I\end{bmatrix} \quad (14)$$
$$\eta=(B\hat{B}^{-1}-I)(u_r+\ddot{x}_{2d})+B\hat{B}^{-1}(f-\hat{f})$$

Assumption 1. $\|E\|=\|B\hat{B}^{-1}-I\|\le\alpha<1$ for some α, for all $x\in R^{2n}$.

Since the inertia matrix M is uniformly positive definite and bounded, there exist positive constants b_1 and b_2 such that

$$b_1\le\|B\|=\|M^{-1}(x)\|\le b_2 \quad (15)$$

for all $x\in R^{2n}$. If we choose

$$\hat{B}^{-1}=\frac{1}{c}I \quad (16)$$

where $c=\frac{b_2+b_1}{2}$, it can be shown that

$$\|E\|=\|B\hat{B}^{-1}-I\|\le\frac{b_2-b_1}{b_2+b_1}=\alpha<1 \quad (17)$$

The point is that there is always at least one choice of $\hat{B}$ satisfying Assumption 1.

Now we design the robust control term u_r. We can find a bounding function $\rho(e,t)$ satisfying the inequalities

$$\|u_r\|<\rho(e,t) \quad (18)$$
$$\|\eta\|<\rho(e,t) \quad (19)$$

From (14),

$$\begin{aligned}\|\eta\| &\le\|E(u_r+\ddot{x}_{2d})+B\hat{B}^{-1}(f-\hat{f})\|\\ &\le\alpha\rho(e,t)+\alpha\|\ddot{x}_{2d}\|+\|B\hat{B}^{-1}\|\cdot\|f-\hat{f}\|\\ &=\rho(e,t)\end{aligned} \quad (20)$$

$$\rho(e,t)=\frac{1}{1-\alpha}[\alpha\|\ddot{x}_{2d}\|+\|B\hat{B}^{-1}\|\cdot\|f-\hat{f}\|] \quad (21)$$

Then, the robust control term u_r is determined to be

$$u_r=\begin{cases}-\rho\dfrac{\rho\Gamma^T P_0 e}{\|\rho\Gamma^T P_0 e\|} & \text{if } \|\rho\Gamma^T P_0 e\|>\varepsilon\\ -\rho\dfrac{\rho\Gamma^T P_0 e}{\varepsilon} & \text{if } \|\rho\Gamma^T P_0 e\|\le\varepsilon\end{cases} \quad (22)$$

where P_0 is the solution to the Lyapunov equation

$$\Lambda^T P_0+P_0\Lambda=-Q_0 \quad (23)$$

and a small positive constant ε is the designer's choice.

Theorem 1. For the uncertain manipulator (1) that can be rewritten in the format (3), controller (12) achieves practical stability in tracking of any reference trajectory x_d, provided x_d satisfies (7).

Proof: We consider a Lyapunov function candidate

$$V=e^T P_0 e \quad (24)$$

Thus

$$\begin{aligned}\dot{V} &= \dot{e}^T P_0 e+e^T P_0\dot{e}\\ &= e^T(\Lambda^T P_0+P_0\Lambda)e+2e^T P_0\Gamma(u_r+\eta)\\ &= -e^T Q_0 e+2e^T P_0\Gamma(u_r+\eta)\end{aligned} \quad (25)$$

If $\|\rho\Gamma^T P_0 e\|>\varepsilon$,

$$\begin{aligned}\dot{V} &= -e^T Q_0 e+2e^T P_0\Gamma(u_r+\eta)\\ &\le -e^T Q_0 e+2\|\Gamma^T P_0 e\|(-\rho+\|\eta\|)<0\end{aligned} \quad (26)$$

If $\|\Gamma^T P_0 e\|\le\varepsilon$,

$$\begin{aligned}\dot{V} &= -e^T Q_0 e+2e^T P_0\Gamma(u_r+\eta)\\ &\le -e^T Q_0 e+2\|\Gamma^T P_0 e\|(-\rho+\|\eta\|)\\ &\le -e^T Q_0 e-2/\varepsilon\|\rho\Gamma^T P_0 e\|^2+2\|\rho\Gamma^T P_0 e\|\\ &\le -e^T Q_0 e+\varepsilon/2\end{aligned} \quad (27)$$

Consequently, according (26) and (27), the error dynamics is practically stable[12,13,14].

4. Conclusions

A new approach for feedback controllers for robot manipulators is proposed. The response of a system is first predicted by appropriate

functional approach and the optimal predictive control law is derived based on the minimization of the local difference between the predicted and desired responses. Then robust control term is added in order to achieve the stability and robustness of the closed-loop systems in presence of uncertainties that arise from imperfect modeling. Stability and robustness analysis of the manipulators with the proposed controller are given.

References

[1] Y. H. Chen, "Robust computed torque schemes of mechanical manipulators: non-adaptive versus adaptive", ASME J. Dynam. Syst. Meas. Contr., Vol. 113, pp. 324-327, 1991.

[2] Y. H. Chen and S. Pandey, "Uncertainty bounded-based hybrid control for robot manipulators", IEEE Trans. Robotics and Automation, Vol. 6, No. 3, pp. 303-311, 1990.

[3] E. Reithmeier and G. Leitmann, "Tracking and force control for a class of robotic manipulators", Dymanics and Control, Vol. 1, pp. 133-150, 1991.

[4] R, Shoureshi, M. Corless, and M. D. Roesler, "Control of industrial manipulators with bounded uncertainties", ASME J. Dynam. Syst. Meas. Contr., Vol. 109, pp. 53-58, 1987.

[5] M. C. Han, "Robust control design for robots with uncertainty and joint-flexibity", J. Korean Society of Precision Engineering, Vol. 12, No. 5, pp. 117-126, 1995.

[6] M. C. Han, "Robust hybrid control for uncertain robot manipulators", J. Korean Society of Precision Engineering, Vol. 14, No. 7, pp. 75-81, 1997.

[7] D. W. Clarke, C. Mohtad, and P. S. Tuffs, " Generalized predictive control - part I. basic algorithm", Automatica, Vol. 14, pp. 413-428, 1978.

[8] B. E. Ydstie, "Extented horizon adaptive control", Proc. 9th IFAC World Congress, Budapest, Hungary, 1984.

[9] A. R. Cauwenberghe, et al., "Self adaptive long range predictive control", Proc. America Control Conference, TP10, pp. 1155-1160, 1985.

[10] P. Lu, "Optimal predictive control of continuous nonlinear systems", Int. J. of Control, Vol. 62, No. 3, pp. 633-649, 1995.

[11] R. Soeterboek, Predictive Control : A Unified Approach, Prentice-Hall, New York, 1992.

[12] G. Leitmann, "On the efficacy of nonlinear control in uncertain linear systems", ASME J. Dynam. Syst. Meas. Contr., Vol. 102, pp. 95-102, 1981.

[13] M. J. Corless and G. Leitmann, "Continuous state feedback guaranteeing uniform ultimate boundedness for uncertain dynamic systems", IEEE Trans. Automat. Contr., Vol. AC-26, No. 5, pp. 1139-1144, 1981.

[14] Y. H. Chen, "Adaptive robust model-following control and application to robot manipulators", ASME J. Dynam. Syst. Meas. Contr., Vol. 109, pp. 209-215, 1987.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Implementation of Virtual Reality Through the Fusion of Visual and Force Information

Sung K. An[1], Seung J. Han[1], Jang M. Lee[2], and Man H. Lee[3]

1. Department of Electronics Engineering, Pusan National University
2. Department of Electronics Engineering & Research Institute of Mechanical Technology, Pusan National University, Keum-Jeung Gu, Jang-Jeon Dong, Pusan, 609-735, Korea
   Tel: +82-51-510-2378, Fax: +82-51-515-5190, E-mail: jmlee@hyowon.cc.pusan.ac.kr
3. School of Mechanical Engineering & ERC/Net Shape & Die Manufacturing, P.N.U

Abstract

A remote virtual reality system is implemented for the operator supervising robot operations at a remote site. For this implementation, a two D.O.F force-reflective joystick is designed to reflect the force/torque measured at the end of robotic manipulator and to generate the motion command for the robot by the operator using this joystick. In addition, the visual information captured by a CCD camera is transmitted to the remote operator and is displayed on a CRT monitor. The operator holding the force reflective joystick and watching the CRT monitor can solve unexpected problems that the robot confronts with. That is, the robot performs the tasks autonomously unless it confronts with unexpected events that can be resolved by only the operator. To demonstrate the feasibility of this system, a remote peg-in-hole operation is implemented and the experimental data are shown.

Keywords: force-reflective, tele-operation, virtual reality

I. Introduction

Robotic manipulators have become increasingly important to the field of flexible automation where operators are supervising the automated task procedure. Also human operators should relinquish the task which are subjected to nuclear weapon, space, underwater, and dangerous places to the robots[1]. Therefore, the robot tasks are not limited to the simple repetitive tasks following a pre-determined trajectory or sequence.

The robots required to perform various tasks in hazardous places may confront unexpected events which can not be resolved by using the knowledge base kept in a robot controller. To overcome the extraordinary situations, human brains need to be included in the control loop. Consequently, the necessity of remote control is emerged recently[2].

For the remote operation, the changes of monitored environment should be closely watched all the time by either a controller or an operator. Conventional industrial robots for repetitive tasks are not applicable any longer to this tele-operation. Therefore, the development of an intelligent robot that actively adapts itself to the changes of working condition has been required. In addition, researches on the virtual-reality based remote control are getting attractive, where the information on the working robot and environment needs to be sent to the remote operator in terms of force and vision information[3].

In this paper, a virtual reality system is implemented through the visual display and force feedback that is transferred to the operator through the two degrees of freedom force-reflective joystick for a remote operator.

II. System Configuration

To measure the force/torque at the end-effector, a wrist force/torque sensor is used. Also, a CCD camera is installed at the working table to capture the information on the robot and working environment, which is sent to the remote operator and displayed on a CRT monitor. To make the operator feel the same force/torque measured at the wrist of the robot and to send the generated commands to the robot, a two degrees of freedom force-reflective joystick is implemented.

F/T and Vision Sensors

A F/T sensor is used to measure the force/torque at the end-effector and attached at the wrist of the robot. The controller for this sensor is manufactured by Assurance Technology (Model: 30/100), which processes the analog data of six strain gage bridges, transforms to a set of six discrete data, and sends the data to the host controller(PC) through a RS232C serial port. The controller sends 13 bytes data set of F/T signals with the baud rate of 38,400 bps to PC within 2.8 *ms*.

To provide the working states and environment of the robot for a remote operator, a CCD camera is attached to the robot end-effector. The visual information captured by the CCD camera is sent to and displayed at the remote CRT monitor for the operator.

Joystick System

To recover the force/torque signals at the end-effector as well as to generate operator's command, a two degrees of freedom joystick is implemented as shown in Fig. 1. The joystick controller is designed using an 80C196KC micro-processor. This retrieves the force at the joystick based upon the force values F_x and F_y, which are sent from PC for the operator. For this purpose, it drives the two axial motors by sending out PWM(Pulse Width Modulation) outputs corresponding to the F_x and F_y. L298N DC motor drives are used to amplify the power.

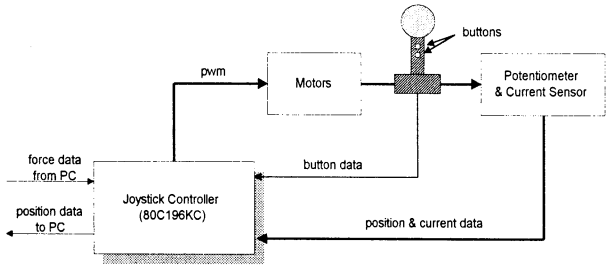


Fig. 1. Block diagram of the joystick system.

Task Robot System

A five axis robot (SCORBOT-ER VII) is used as shown in the left side of Fig. 2. The robot consists of five axes: Base (1st axis), Shoulder (2nd axis), Elbow (3rd axis), Pitch (4th axis), and Roll (5th axis). The position coordinates of the robot are represented by the joint angles of the 1st, 2nd, 3rd, and 4th axes; the orientational coordinates are represented by the joint angles of the 4th and 5th axes. The robot controller uses a Motorola 68020 as a CPU. The PID algorithm is applied to the position control of electrical DC servo motors and its repetition accuracy is 0.2 mm. The robot weighs 30 kg and it communicates with the host controller through a RS232C serial port.

Teleoperated System

A teleoperated robot system is implemented as in Fig. 2 by integrating a F/T sensor, a vision sensor, a joystick system, and a robot system. In the autonomous mode of the robot, the controller generates commands based upon the pre-programmed algorithm stored in the host controller utilizing the F/T sensor data. The remote operator feels like he is working at the local task site with the aids of the visual feedback and force reflection, and generates motion commands for the robot using the joystick. The main mission of the supervising operator is to resolve the unexpected situations for the autonomous robot by sending proper commands.

III. Control of a Teleoperation System

The control of teleoperation system is divided into two categories: a remote joystick control and a task robot control. The role of the joystick is reflecting the F/T values to the operator as well as generating the robot position commands for the task

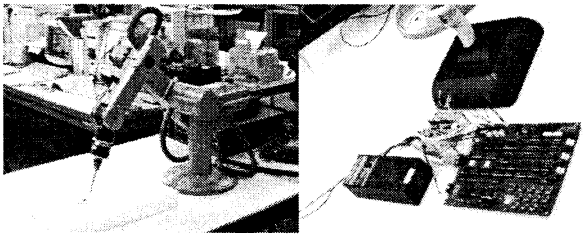


Fig. 2. Teleoperation system.

robot. The task robot performs the given tasks autonomously based upon the pre-programmed algorithm using the measured force/torque values and the pre-specified position commands. It also accepts the position commands sent from the operator with the highest priority. In this section, the control systems of remote joystick and the task robot are described.

3.1 Control of Joystick System

The force exerted at the end-effector, F_t, can be measured as the output of the wrist force/torque sensor: F_x, F_y, F_z, T_x, T_y, and T_z. Since two degrees of freedom joystick is used in this research, only F_x and F_y components of F_t are used in reflecting the force at the joystick. F_x represented in the F/T sensor coordinate is transformed into the up/down directional force J_x, of the joystick while F_y is transformed into the left/right directional force J_y. The transformations are defined as Eqs. (1) and (2). Here k_f is a force gain constant. Also the transformation between the position in joystick coordinate and the joint angle of the robot is defined as Eqs. (3) ~ (6). In these equations, Δx, Δy, Δz, and Δp are x-axis, y-axis, z-axis, and pitch variations of the robot, respectively. And k_t is a gain constant for transforming the force to positional variations.

$$J_x = k_f \cdot F_x \tag{1}$$
$$J_y = k_f \cdot F_y \tag{2}$$
$$\Delta x = k_t \cdot (J_x \cdot \cos(a) + J_y \cdot \cos(a - 90^\circ)) \tag{3}$$
$$\Delta y = k_t \cdot (J_x \cdot \sin(a) + J_y \cdot \sin(a - 90^\circ)) \tag{4}$$
$$\Delta p = k_t \cdot J_x \tag{5}$$
$$\Delta z = k_t \cdot J_x \tag{6}$$

The control block diagram of the joystick is shown in Fig. 3. Here the force inserted by the operator to the joystick, F_h, is treated as disturbance, which can be estimated by measuring the current flowing through the motor windings. The transfer function from the operator's force, F_h, to the motor current, I_a, is obtained as Eq. (7). It can be also rearranged to represent the operator's force, F_h, as Eq. (8).

$$I_a(s) = \frac{E_a(s)}{RJs + BR + KK_b} + \frac{K_b F_h(s)}{RJs + BR + KK_b} \tag{7}$$
$$F_h(s) = \frac{E_a(s)}{K_b} + \frac{(RJs + BR + KK_b) I_a(s)}{K_b} \tag{8}$$

The contact force F_x and F_y are sent to the 80C196KC micro-controller and transformed to J_x and J_y for the joystick control. J_x governs the up/down motion of the joystick; J_y the left/right motion. Note that the motion or positional variation of the joystick held by the operator generates the reflecting force.

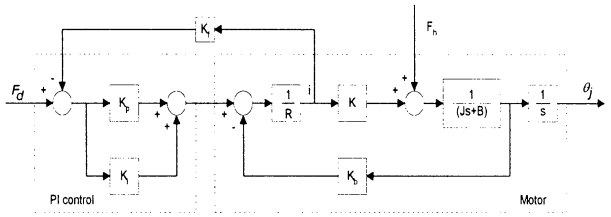


Fig. 3. Block diagram of joystick control system.

3.2 Control of Task Robot

The robot performs the peg-in-hole operation according to the pre-programmed commands in the autonomous mode, as shown in Fig. 4. When the peg arrives at P_2, the robot changes the pitch of the peg until it becomes vertical to the hole. Then the peg-in-hole task is performed by moving the peg into P_3. During this task, unless the operator is generating a new command, the robot is continuously operating in the autonomous mode.

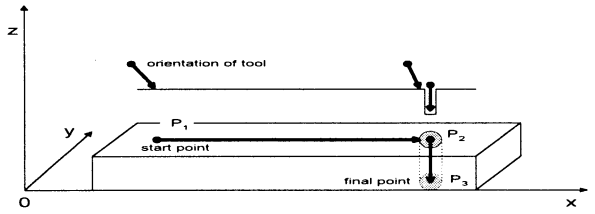


Fig. 4. Experimental environment.

For the surface following operation (from P_1 to P_2), a position-based impedance control is applied to maintain a constant contacting force[4,5]. The desired impedance is specified as follows:

$$M\ddot{E}+B\dot{E}+KE=F_e \tag{9}$$

where, M, B, and K are $n \times n$ positive definite diagonal matrices, $E = X_r - X$, and X_r is an $n \times 1$ reference trajectory.
Also, the contact force, F_e, is defined as

$$F_e=K_e(X-X_e) \tag{10}$$

where, K_e is a diagonal stiffness matrix and X_e is an $n \times 1$ position vector.

Here, in this paper, z directional force control, f_r, is aimed to be maintained constant. To achieve constant, f_r, the measured force, f_e, needs to be controlled actually, which is possible by adjusting a reference trajectory, x_r. The reference trajectory, x_r, can be represented in terms of f_r, x_e, k and k_e as Eq. (11). Note that M and B matrices in Eq. (9) are set to be null.

$$x_r=x_e+f_r \cdot \frac{k+k_e}{kk_e} . \tag{11}$$

In reality, the stiffness of environment, k_e, is unknown. However it can be estimated using the measured force and position information as follows:

$$k_e=\frac{f_e}{x-x_e} . \tag{12}$$

Now the reference trajectory, x_r, is obtained by substituting Eq. (12) into Eq. (11) as Eq. (13).

$$x_r=x_e+(x-x_e) \cdot \frac{f_r}{f_e}+\frac{f_r}{k} . \tag{13}$$

This reference trajectory is fed to the robot controller to satisfy the desired impedance during the surface tracking operation.

Ⅳ. Experiments

The peg-in-hole task is implemented by a robot that task is monitored by a remote operator as shown in Fig. 4 and Fig. 2.

4.1 Autonomous Task Execution

Fig. 5 represents the x-directional forces measured at the end-effector of the robot operating in the autonomous mode. The force values are transmitted to and calibrated by the joystick controller within a bound of $-12.5\,N \sim +12.5\,N$ with an offset bias of $7.2\,N$ to make the operator comfortable in feeling the reflected force. As the results of experiment, it is formal that the reflected force is well following the measured force except the insertion period. During the insertion, the measured force becomes large and fluctuating since the peg is contacting the hole surface in a irregularly way.

Fig. 6 shows the changes of z and p during the peg-in-hole operation. In the operation, the robot moves along the x axis from P_1 to P_2 until $T=200$ sec, changes its pitch until $T=560$ sec, and inserts the peg into the hole until $T=590$ sec. Note that at $T=590$ sec, the robot stops the insertion before it reaches the final point P_3, since the force F_x at the end-effector exceed the driving capability of the robot.

4.2 Supervisory Task Execution

The robot basically operates in the autonomous mode. However it can not keep the pre-programmed commands when unexpected events occur. The operator grasping a force reflective joystick can perceive the situations by feeling the force at the joystick. At this moment, the operator generates new position commands for the robot using the joystick. Note that in many cases, the insertion can not be performed without the aids of the operator.

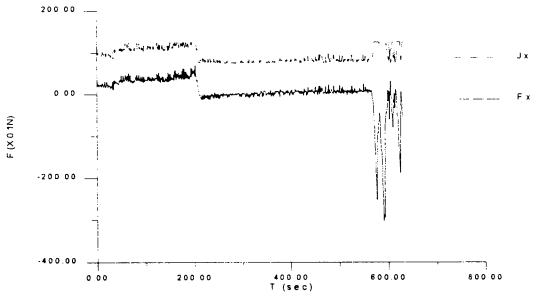


Fig. 5. F_x tracking in autonomous mode.

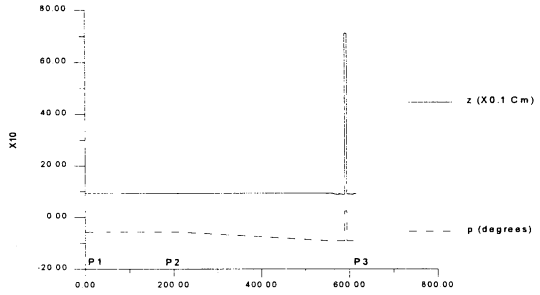


Fig. 6. Position tracking in autonomous mode.

Fig. 7 shows F_x, and the calibrated force at the joystick, J_x in the supervisory mode. Here, it is noticed that the force of joystick faithfully follows the force of robot end-effector. Fig. 8 shows that the robot keeps its desired trajectory correctly like in the autonomous mode and it stops at $T=590$ sec. At this moment, the operator adjusts the peg so that the robot can move the peg into the hole until it reaches to P_3 successfully.

4.3 Impedance control from P1 to P2

Fig. 9 shows the contacting force along z axis while the peg is moving from P_1 to P_2 in the autonomous mode. The reference trajectory for this position based impedance control is generated according to Eq. (11). During this impedance control, the x directional velocity is kept constant as $0.01\ cm/cycle$ not to change the total task execution time. The z directional reference force is assigned as $-27\ N$, and the control cycle is $140\ msec$. Note that the desired contacting force is well kept along the trajectory using this impedance control.

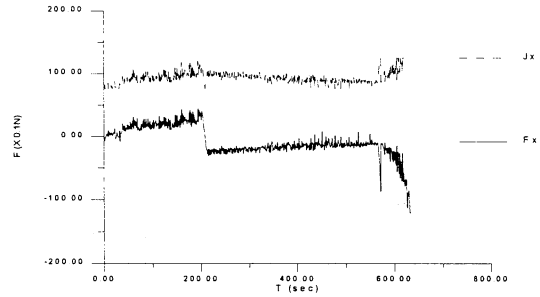


Fig. 7. F_x tracking in supervisory mode.

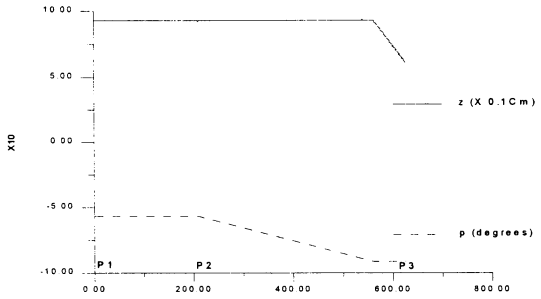


Fig. 8. Position tracking in supervisory mode.

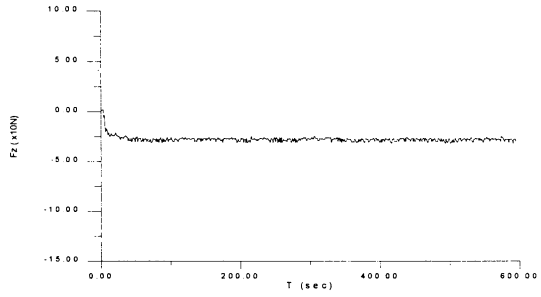


Fig. 9. Impedance control of robot from P1 to P2.

V. Conclusion

A virtual reality system is implemented for an operator supervising a robot's task at a remote site. A peg-in-hole operation is implemented to show the effectiveness of this system in resolving the jamming of the peg in the hole. For the operator's appropriate decision, a two degrees of freedom force reflective joystick is implemented to reflect the actual force measured at the wrist of the robot.

Acknowledgement

This work was supported by ERC/Net Shape & Die Manufacturing, and the Research Fund of the POSCO Chair Professor at Pusan National University.

References

[1] Vibet C. (1986), "Emerging Methods in Master-Slave Manipulator Design," Int. J. of Robotics and Automation, vol. 1, no. 2, pp. 58-62

[2] Sheridan, T. B. (1989), "Telerobotics", Automatica, vol. 25, no. 4

[3] D. H. CHA and H. S. CHO (1996), "Design of a Force Reflection Controller for Telerobot Systems Using Neural Network and Fuzzy Logic," J. of Intelligent and Robotic Systems, pp. 1-24

[4] N. Hogan (1985), "Impedance control : An approach to manipulatoi," ASME Journal of Dynamic Systems, Measurement, and Control, vol. 3, pp. 1-24

[5] S. Jung and T.C. Hsia and R. G. Bonitz (1995), "On Force Tracking Impedance Control with Unknown Environment Stiffness," IASTED Conference on Robotics and Manufacturing, June, pp. 181-184

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Hardware Implementation of Fuzzy Logic Controller for Speed of a DC Series Motor Using an Adaptive Evolutionary Computation

Gi-Hyun Hwang * Kyeong-Jun Mun* J. H. Park* M. H. Lee***

* Department of Electrical Engineering, Pusan National University, Pusan, 609-735, Korea
** School of Mechanical Engineering & ERC/Net Shape & Die Manufacturing, Pusan National University, Pusan, 609-735, Korea

Abstract

In this paper, an Adaptive Evolutionary Computation (AEC) is proposed. AEC uses a genetic algorithm (GA) and an evolution strategy (ES) in an adaptive manner in order to take merits of two different evolutionary computations: global search capability of GA and local search capability of ES. In the reproduction procedure, proportions of the population by GA and the population by ES are adaptively modulated according to the fitness. AEC is used to design of the membership functions and scaling factors of fuzzy logic controller (FLC).

To evaluate the performances of the proposed FLC, we make an experiment on FLC for the speed control of an actual DC series motor system with nonlinear characteristics. Experimental results show that proposed controller have better performance than those of PD controller.

I. INTRODUCTION

During the last decade, fuzzy logic control has attracted great attention from both the academic and industrial communities. Recently, fuzzy logic controller has been suggested as an alternative approach to conventional control techniques for complex control system, such as nonlinear or time delay system. That is, the design of fuzzy logic controller (FLC) does not require a mathematical description of the control system and the fuzzy controller can compensate the environmental variation during operating process [1-3].

However, we cannot obtain good control performances if the membership functions, fuzzy rules and scaling factors are incorrect. Recently, the membership functions, fuzzy rules and scaling factors are determined by evolutionary computations (ECs), which is the probabilistic search method based on genetics and evolutionary theory [4, 5].

ECs are optimization algorithms based on the principles of the genetics and natural selection. There are three broadly similar avenues of investigation in ECs: genetic algorithm (GA), evolution strategy (ES), and evolutionary programming (EP) [4-6]. When applied for practical problem solving, each begins with a population of contending trial solutions brought to the task at hand. New solutions are created by randomly altering the existing solutions by EC operation. An objective measure of performance is used to assess the fitness of each trial solution and selection.

It is obvious from the start that finding good settings for EC parameters for a particular problem is not a trivial task. Several approaches were proposed. One approach uses adapting population size, crossover rate, and mutation rate. Arabas [7] proposed an adaptive method for maintaining variable population size. Schlierkamp-Voosen [8] present a competition scheme, which dynamically allocates the number of trials given to different search strategies. The competition scheme changes not only the sizes of the subgroups, but also the size of the whole population. Srinivas [9] proposed the Adaptive Genetic Algorithm (AGA), that is, the probabilities of crossover and mutation are varied depending on the fitness values of the solutions to maintain diversity in the population and to sustain the convergence capacity of the GA.

The other approach involves 1) adapting the probabilities of crossover and mutation operator in GA: the idea is that the probability of applying an operator is altered in proportional to the observed performance of the individual created by this operator. 2) Mutation parameters are adapted during the run in ES. Hinterding [10] proposed Gaussian mutation operators for GA, which allows the GA to vary the mutation strength during the run. Spears [11] proposed an adaptive mechanism for controlling the use of crossover in an ECs and explores the behavior of this mechanism in a number of different situations.

In this paper, a new methodology of evolutionary computations - an Adaptive Evolutionary Computation (AEC) - is proposed. AEC uses a GA and an ES in an adaptive manner in order to take merits of two different evolutionary computations: global search capability of GA and local search capability of ES. In the reproduction procedure, proportions of the population by GA and the population by ES are adaptively modulated according to the fitness. AEC is used to design of the membership functions and scaling factors of FLC. The proposed FLC is applied to the speed control of an actual DC series motor system with nonlinear characteristics.

II. ADAPTIVE EVOLUTIONARY COMPUTATION

A. Motivation

In general, GA is known to offer significant advantages over traditional optimization methods. The most important ones are: a population-based search, a balance between exploitation (convergence) and exploration (diversity). But GA can suffer from excessively slow convergence before providing an accurate solution because of their not exploiting local information. On the other hand, ES is well known to exploit all local information in an efficient way. But, for problems with many local minima, it has the possibility of trapping in local minima.

In this paper, to reach accurately and reliably the global optimum in a short execution time, we designed AEC bringing together the benefits of GA and ES. In the AEC, GA operators and ES operators are simultaneously applied to the individuals of the present generation to create next generation. Individuals with higher fitness value will have a higher probability of contributing one or more chromosomes in the next generation. This mechanism should give greater reward to either the GA operation or the ES operation that produces superior offspring.

B. Adaptive Evolutionary Computation

In the AEC, the number of individuals created by GA operation and the number of individuals created by ES operation change adaptively. Configuration of the AEC is shown in Fig. 1. In AEC, the individual is represented as a real number chromosome, not a binary chromosome, which makes it possible to hybridize for GA operation and ES operation without loss of data. The main objective behind such implementation is that it enhances the performances of AEC.

ES forms a class of optimization techniques motivated by the reproduction of biological system and a population of individuals evolves toward better solutions by means of the mutation and selection operation. In this paper, we adopted (μ, λ)-ES, that is, only the λ offspring by mutation operation compete for survival and the μ parents are completely replaced each generation. Also, self-adaptive mutation step sizes are used in ES.

For the AEC to self-adapt its use of GA and ES, each individual has an operator code in order to represent whether governed by GA or ES. Suppose a '0' refers to GA, and a '1' to ES. At each generation, if it is more proper to use GA, more '0's should appear in the end of individuals. If it is more proper to use ES, more '1's should appear. After reproduction by roulette wheel selection according to the fitness, GA operations, that is, crossover and mutation are performed on the individuals of which operator code is '0' and ES operation, that is, mutation is performed on the individuals of which operator code is '1'. Elitism is also used. Best individual in the population is preserved to perform both GA operations and ES operation to next generation.

III. DESIGN OF FUZZY LOGIC CONTROLLER USING ADAPTIVE EVOLUTIONARY COMPUTATION

In designing a FLC, the exact mathematical modeling of control system is not needed and fuzzy rules can be represented as the knowledge of the experts. The design parameters used in this paper are given below.

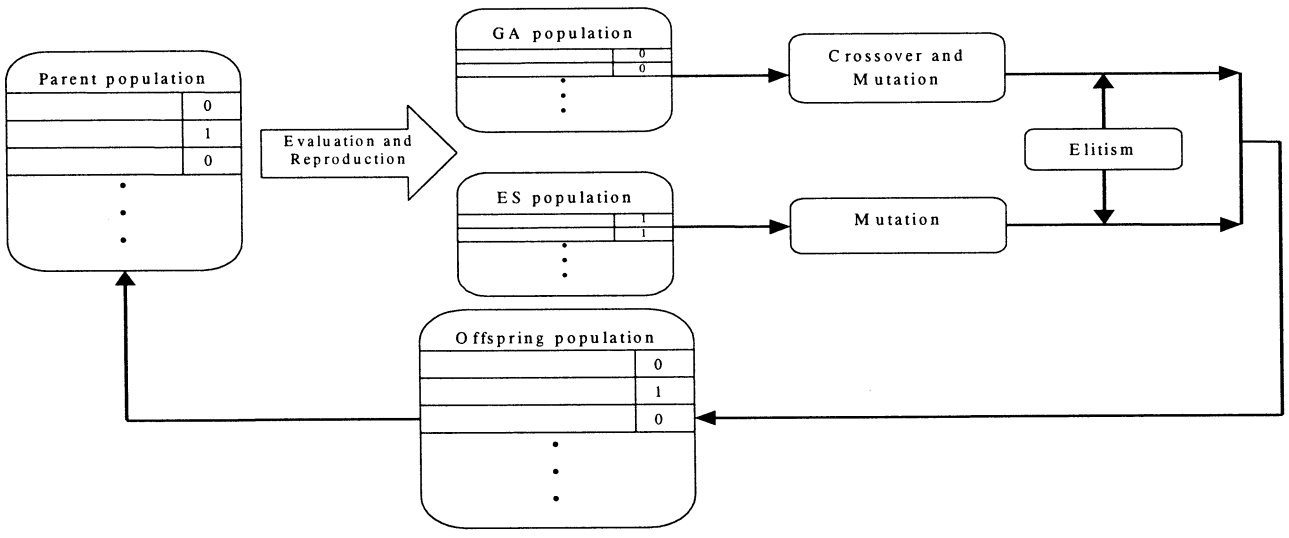


Fig. 1 Configuration of Adaptive Evolutionary Computation

- number of input/output variables : 2/1
- number of input/output membership functions : 7/7
- fuzzy inference method : max-min method
- defuzzification method : center of gravity

In this paper, the proposed method is to optimize the shapes of the membership functions and scaling factors by the AEC. The general scheme is presented in Fig. 2. The input signals to the FLC are speed deviation (e) and the change in speed error (de). The output signals of the FLC are used to the speed control of an actual DC series motor system. Because error and change-of-error is used as input variables of the FLC, PD-like FLC is used. Rule base for the PD-like FLC from the two-dimensional phase plane of the system in terms of error and change-of-error is shown in Table 1. The general approach to design the FLC is the division of the phase plane into two semi-planes, by means of switching-line. Within the semi-planes positive and negative control outputs are produced, respectively.

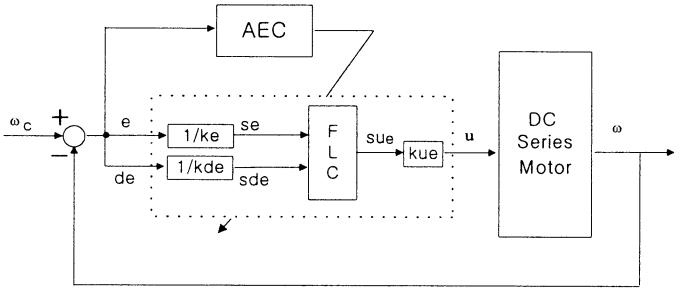


Fig. 2 Block diagram of fuzzy logic controller using the AEC

Table 1 PD-type fuzzy rules

e \ de	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NB	NM	NM	NS	ZE
NM	NB	NB	NM	NM	NS	ZE	PS
NS	NB	NM	NM	NS	ZE	PS	PM
ZE	NM	NM	NS	ZE	PS	PM	PM
PS	NM	NS	ZE	PS	PM	PM	PB
PM	NS	ZE	PS	PM	PM	PB	PB
PB	ZE	PS	PM	PM	PB	PB	PB

switching line

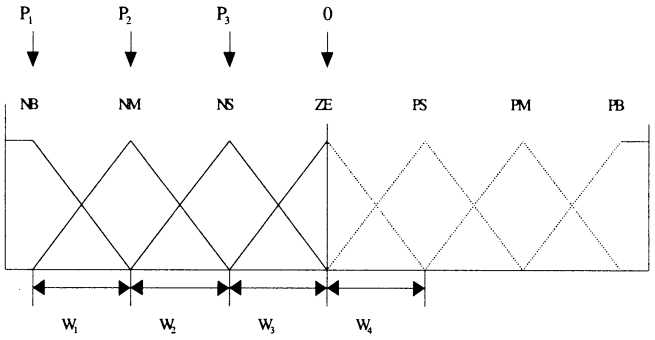
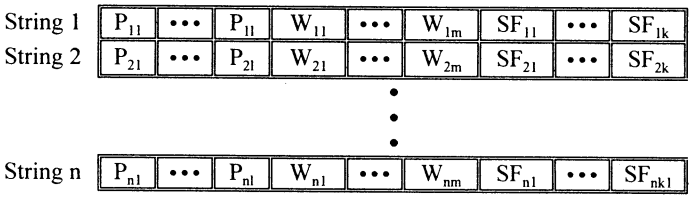


Fig. 3 Symmetrical membership functions

The magnitude of the output signals depends on the distance of the state vector from the switching line. When tuning the membership functions by the AEC, fuzzy rules are symmetric about switching line as shown in Table 1. Where, linguistic variable NB means "Negative Big", NM means "Negative Medium", NS means "Negative Small", ZE means "Zero", PS means "Positive Small", PM means "Positive Medium", and PB means "Positive Big". The shape of the input and output membership function is assumed to be triangular. Also we use 7 input/output fuzzy sets for every input/output variable, hence the number of parameters of the FLC (center and width of the membership functions) is 63. But it takes long time for the AEC to tune 63 fuzzy parameters. In this paper, we set the ZE membership function to 0 and positive and negative membership function is symmetric about the 0. So the number of parameters of the FLC is 21, that is, 3 centers and 4 width for each variable, as shown in Fig. 3. Also scaling factors of the FLC are tuned using the AEC, as shown in Fig. 2. To encode each parameter, real coding technique is used. String architecture for tuning the membership functions and scaling factors is shown in Fig. 4. To evaluate each string in the population, the absolute error between output speed and reference speed of generator is used. The fitness function is defined in (1).

$$Fitness = \frac{1}{100 + \sum_{k=1}^{N} |\omega_{ref} - \omega_k|} \quad (1)$$

where, ω_k : actual speed
ω_c : desired speed
N : no. of data acquired during T second



Where, n : population size
l : No. of center of the membership functions
m : No. of width of the membership functions
k : No. of scaling factors

Fig. 4 Strings architecture for tuning membership functions and scaling factors

IV. EXPERIMENTAL RESULTS

Fig. 5 shows speed control system structure of the speed control of an actual DC series motor. As shown in Fig. 5, the AEC is used to optimize the shapes of the membership functions and scaling factors. Table 2 shows the simulation parameters of the AEC for tuning the FLC. Fig. 6 shows the shape of the membership functions by the AEC.

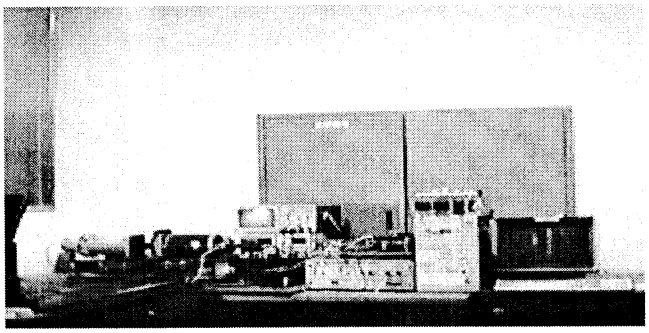
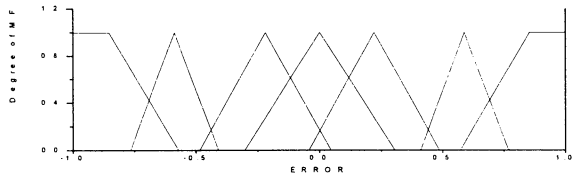


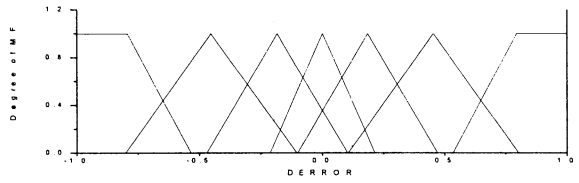
Fig. 5 Experiment apparatus of a DC series motor system

Table II SIMULATION PARAMETERS USED AEC

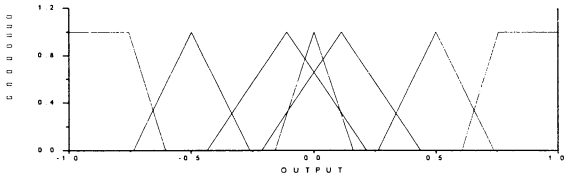
Methods	AEC
Size of population	20
Crossover probability	0.85
Mutation probability	0.05
δ	0.5
C_d	0.85
C_1	1.15
Number of Generation	20



(a)The membership function of error



(b) The membership function of error rate



(c)The membership function of output

Fig. 6 Tuned membership functions using AEC

Fig. 7 (a) provides graph of the fitness values by the AEC. Fig. 7 (b) provides graphs of the number of individuals for GA operations and ES operation in the AEC. As shown in Fig.7, the percentage of individuals for GA operation is greater than that of individuals for ES operation in initial generation. But, from generation to generation, the percentage of individuals for ES operation exceeds that of individuals for GA operation. The AEC produces improved reliability by exploiting the "global" nature of the GA initially as well as the "local" improvement capabilities of the ES from generation to generation.

Fig. 8 represent experimental results of the DC series motor system for reference command used when tuning the FLC and the PD controller. As shown in Fig. 8, speed response of PD controller produces many differences between desired speed (ω_c) and actual speed (ω). Whereas, proposed FLC produces more accurate speed response than PD controller in terns of tracking performance. Therefore, the proposed FLC demonstrates a better tracking performance as compared with the PD controller. To evaluate the robustness of the FLC, FLC is also tested over the reference command, which is not used when tuning. As shown in Fig. 9, experimental results confirm that FLC shows better performance over another reference command and various disturbances than that of PD controller.

V. CONCLUSIONS

In this paper, we have adaptively coupled GA with ES. The reason for combining GA with ES is that they compliment each other. ES will try to optimize locally, while the GA will try to optimize globally. In the AEC, GA operators and ES operators are simultaneously applied to individuals of the present generation to create next generation. In the AEC, the number of individuals created by GA operation and the number of individuals created by ES operation change adaptively. The AEC produces improved reliability by exploiting the "global" nature of the GA initially as well as the "local" improvement capabilities of the ES from generation to generation, so the AEC converges to the global optimal solution within a few generations.

The AEC is used to design of the membership functions and scaling factors of FLC. The proposed FLC is applied to the speed control of an actual DC series motor system with nonlinear characteristics. Experimental results show that FLC has better control performance than PD controller in terms of rising time, settling time. Also FLC has better performance over another reference command.

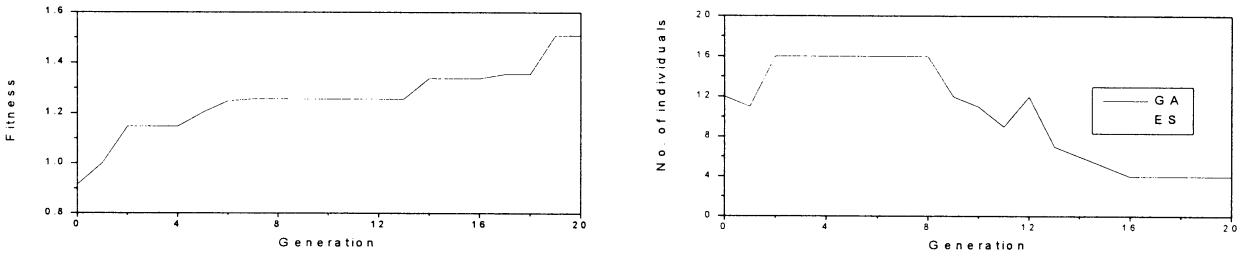
ACKNOWLEDGEMENT

This work was supported by the research fund of Pusan National University.

REFERENCES

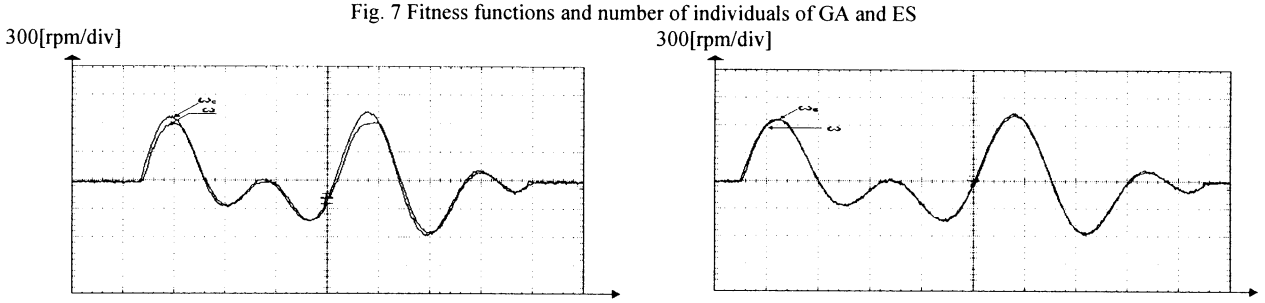
[1] Li-Xin Wang, "Stable Adaptive Fuzzy Controllers with Application to Inverted Pendulum Tracking", IEEE Trans. On Systems, Man, and Cybernetics-Part B : Cybernatics, Vol. 26, No. 5, pp. 677-691, Oct. 1996

[2] Abraham Kandel, Gideon Langholz, "Fuzzy Control Systems", CRC Press, 1994Y. S. Kung and C. M. Liaw, "A Fuzzy Controller Improving a Linear Model



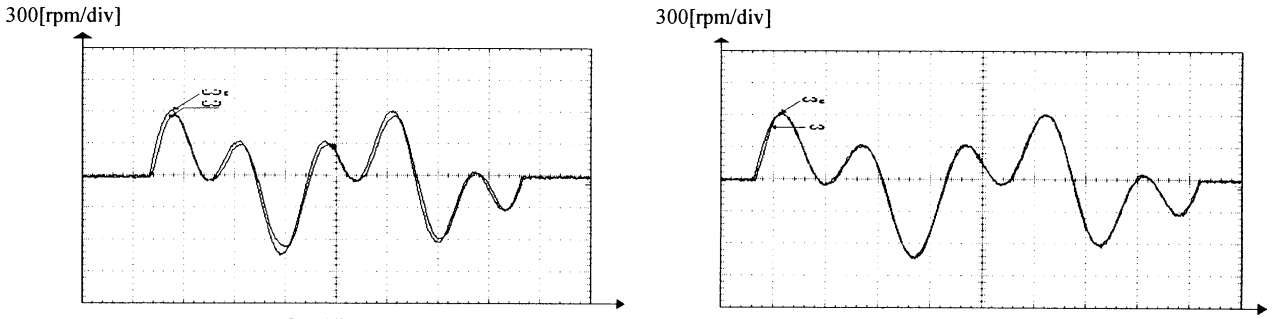
(a) Fitness value (b) Number of individuals of GA and ES in AEC

Fig. 7 Fitness functions and number of individuals of GA and ES



(a) Speed response (PD Controller) (b) Speed response (FLC)

Fig. 8 Comparisons of speed response with the PD controller and fuzzy logic controller



(a) PD controller (b) Fuzzy logic controller

Fig. 9 Speed response with new reference speed

Following Controller for Motor Drives", IEEE Trans. On Fuzzy Systems, Vol. 2, No. 3, pp. 194-201, Aug., 1994

[3] D. E. Goldberg, Genetic Algorithms in Search optimization, and Machine Learning, Addison-Wesley publishing Company, INC., 1989

[4] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag, 1994

[5] Mitsuo Gen and R. Cheng, Genetic Algorithms & Engineering Design, A Wiley-Interscience Publication, 1997

[6] J. Arabas, Z. Michalewicz, and J. Mulawka, "GAVaPS-a Genetic Algorithm with Varying Population Size", IEEE International Conference on Evolutionary Computation, pp. 73-78, 1994

[7] D. Schlierkamp-Voosen and H. Muhlenbein, "Adaptation of Population Sizes by Competing Subpopulations", IEEE International Conference on Evolutionary Computation, pp. 330-335, 1996

[8] M. Srinivas and L. M. Patnaik, "Adaptive probabilities of Crossover and Mutation in Genetic Algorithms", IEEE Trans. on Systems, Man and Cybernetics, Vol. 24, No. 4, pp.656-667, April, 1994

[9] Robert Hinterding, "Gaussian Mutation and Self-adaptation for Numeric Genetic Algorithms", IEEE International Conference on Evolutionary Computation, pp. 384-389, 1995

[10] W. M. Spears, Evolutionary Programming □ □The MIT Press, 1995

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Development of a 3D Graphic Simulation Tool for a SCARA Robot

D. Y. Lee, J. W. Choi, M. H. Lee, K. Son, M. C. Lee
School of Mechanical Engineering and RIMT
Pusan National University, Pusan 609-735, Korea

J. M. Lee
Department of Electronics Engineering
Pusan National University, Pusan 609-735, Korea

S. H. Han
Department of Mechanical Engineering
Kyungnam University, Masan 631-701

Abstract

In this paper, we developed a Windows 95 version Off-Line Programming System which can simulate a Robot model in a 3D graphic space. 4 axes SCARA robot (especially FARA SM5) is adopted as a test model. Forward kinematics, inverse kinematics and robot dynamics modeling were included in the developed program. The interface between users and the OLP system in the Windows 95's GUI environment was also studied. The developing language is Microsoft Visual C++. The graphic libraries(OpenGL) which are provided by Silicon Graphics, Inc. were utilized for the tool.

1. Introduction

In recent years, laborer's high fare and consumers' request of multi-product small-batch productions are very difficult problem to be satisfied. As a solution for the problem, factory automation had been considered and in progress by using robots in work line, and has been leaded by many major enterprises. The line automation is expected to improve producing rate and quality. In general, task teaching and modified system evaluations are accomplished by on-line method for the current systems. In this method, another developing line and well-trained experts are needed to teach a new task and to evaluate the results of performance experiment for the new task. In addition, the teaching should be repeated many times for more complicated tasks even for the small change of the system.

These problems could be solved by employing an Off-Line Programing(OLP) System which has the similar environment of the real world. The OLP Systems are defined as a robot programming language which has been sufficiently extended, generally by means of computer graphics, that the development of robot programs can take place without access to the robot itself[1]. Because of this reason, by using the OLP System, dynamic simulations of a robot could be available without the real robot operations. In addition, an easy evaluation and development test for the teaching, trajectory planning, control algorithms could be accomplished just in software environment. Therefore, work stops of production line are eliminated by using Off-Line teaching and Off-Line performance evaluation[1][2].

There are several previously developed OLP Systems such that ROSI2, STAR, SILMA Inc.'s Cimstation, Robot Simulation Ltd.'s WORKSPACE, BYG system Ltd.'s GRASP, Technomatrix Technologies Ltd.'s ROBOCAD, and Denob Robotics Inc.'s IGRIP[3]. At present, most OLP Systems provide GUI environment., but the OLP Systems require workstation level computers. Because of this reason, such kind of OLP Systems are not widely used.

In this paper, we developed a Windows 95 version Off-Line Programming System which can simulate a robot model in a 3D graphic space. 4 axes SCARA robot (especially FARA SM5) is adopted as a test model. Forward kinematics, inverse kinematics and robot dynamics modeling were included in the developed program. The interface between users and the OLP System in the Windows 95's GUI environment was also studied. The developing language is Microsoft Visual C++. The graphic libraries[4](OpenGL) which are provided by Silicon Graphics, Inc. were utilized for the tool[5].

2. Basic Element of the General OLP System and Solution Method of the Developed Program

In the developing process of the OLP System, Bolles and Roth[1] referred about basic 5 elements as a necessary condition. Of course, other various elements may be exist, and may also be considered in the development. In this paper, the process of the program development is focused on Bolles and Roth's the five basic elements[1][2].

2.1 User interface

In the developing process of an OLP System, user's usage and data management have to be easy and comfortable to use. Thus, the user interface method must be exploited for the substitution of robot teach pendant[1][2].

In the process, the former suggested problems are solved with upgrading programs for Windows 95 version. Because

Windows 95 has GUI as default environment for the user convenience[4]. By using and modifying various dialog boxes, shortcut key and mouse use benefit which are provided by the OS, we are focused on easy usage for the OLP System development under the Windows 95 environment.

2.2 3D modeling

A central element in the OLP System is the use of graphic depictions of the simulated robot and its workcell. This requires the robot and all fixtures, parts, and tools in the workcell to be modeled as three-dimensional objects. With this ability, 3D modeled objects in the monitor plane must have the animation ability[1][2].

This can be solved by constructing 3D modeling with Silion Graphics Inc.'s graphic library OpenGL. The library enables the program to animate 3D model faster and to remove some blinks of high precision animation in the use of Windows 95's default GDI[5].

2.3 Kinematic emulation

A central component in maintaining the validity of the simulated world is the faithful emulation of the geometrical aspects of each simulated manipulator. Mainly, this problem is solved by including robot's forward kinematics and inverse kinematics in the program[1][2].

Because the SCARA robot, FARA SM5's forward, inverse kinematics are not so difficult to solve, we can develop the program by solving equation and including these subroutines in the program. Additionally, parameter changes of a real robot such that link length, weight change, are also considered by user parameter changing function.

2.4 Path planning emulation

An OLP System should accurately emulate the path taken by the manipulator in moving through space. This emulation of the spatial shape of the path taken is important for detection of collisions between the robot and its environment, and for robot task teaching[1][2].

For the problem, the developed program enables the user to choose one among five path planning methods which are also developed and provided in the program. Virtual teach pendant is constructed with Windows 95's dialog box. It also provides path planning capability.

2.5 Dynamic emulation

Simulated motion of manipulators can neglect dynamic attributes, if the OLP System performs a good job of emulating the trajectory planning algorithm of a controller, and if the actual robot follows desired trajectories with negligible errors. However, at high speed or under heavy loading conditions, the trajectory tracking error may be important. Simulations for investigating these tracking errors necessitate dynamics modeling of the manipulator and objects[1][2].

For this problem, we derived the dynamic equations for the SCARA robot using Lagrange-Euler formula about non-conservative system and developed subroutines by encoding them with C++ language. The derived dynamic equations also include the DC servo motor dynamics.

3. Structure and Function of the Developed OLP System

3.1 Whole structure of the OLP System

Fig. 1 depicts the whole structure of the OLP System. The OLP System is divided into 3 parts, that is, *Setting and Teaching*, *Simulation*, and *Evaluation* parts. In *Setting and Teaching* part, parameter setting, teaching via points, trajectory generating can be done. In *Simulation* part, by the given control algorithm, dynamic simulations can be accomplished. Finally, in *Evaluation* part, the simulated data are evaluated by the 3D animation, and the tracking performance of the employed control algorithms are also evaluated.

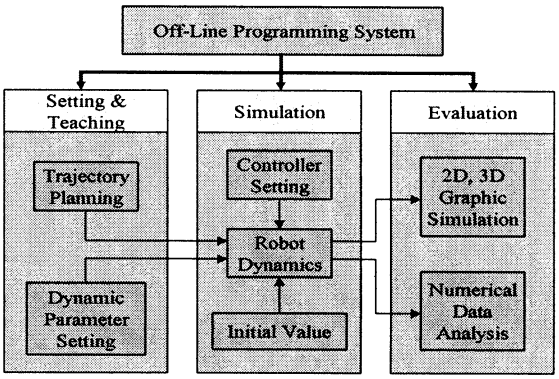


Fig. 1 Structure of the OLP System

3.2 Initial view of the OLP System

At program starting(Fig. 2), a basic frame which has the pulldown type menu is browsed, and the SCARA robot is drawn in this initial window's client area. The SCARA robot pose is default posture whose all joint values are zero.

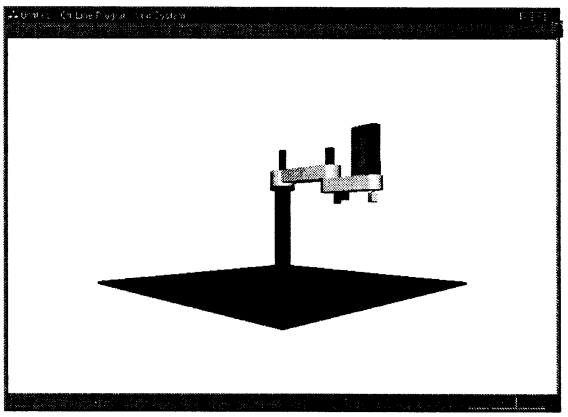


Fig. 2 Initial Window of OLP System

3.3 Structure of the OLP System menu

Table 1 shows the pulldown menu of the developed OLP System. Execution menu of the table is not perfectly constructed because it provides the connection of a real robot and the OLP System. Studies on this function are in process.

3.3.1 File menu and management

The OLP System manages all information such that via

Table 1. Menu Structure of OLP System

FILE	SETUP	TEACHING	SIMULATION	EXECUTION	VIEW	HELP
New	Specification	New	Control Type	On-Line	Change Posture	About OLPS
Open	Work Range	Open	Run	Verification	Change View	
Save..	Max. Velocity	Trajectory	Evaluation	Comparison	Wire Frame	
Save as	Load					
Exit						

point information, standard trajectory data, simulated trajectory data, as a data file format. Each file has the inherent extension which are given by the developer. A new data file is created or existing files are opened before a task is started. Fig. 3 shows the file open scene using *Open* command in the pulldown menu.

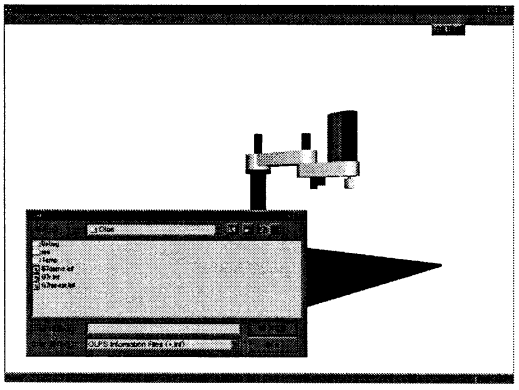


Fig. 3 Execute Open Command in FILE Menu

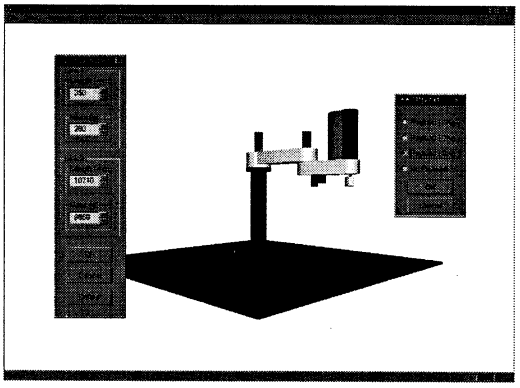


Fig. 4 Specification & Payload Dialog Box of Setup Menu

3.3.2 Setup menu

In Setup menu, each length of the links, weight, work range, maximum velocity and payload can be assigned before tasks are started. In Specification menu, users can assign link length for mm scale, link weight for gram scale. In Work range menu, users can assign maximum range of each joint, and in Max. Velocity, maximum velocity of each joint can also be assigned. These assigned values are used for limit conditions in the trajectory planning. In each menu, users can change the given data with the keyboard and mouse. Fig. 4 shows the scene that data are changed in dialog box.

3.3.3 Teaching menu and trajectory planning

Teachings are accomplished after a teaching data file is created, and this file contains various teaching information. This teaching data file is created through dialog box, and various teaching data are inputted and saved in the memory. The necessary teaching information is the via point in cartesian coordinate or in joint coordinate, passing time, and the trajectory planning method between via points. Via point input in joint coordinate is accomplished by inputting 4 joint angle values, and in cartesian coordinates by inputting x, y, z values and rotating angles of end effector. Passing time is inputted by second scale, and cubic spline method, LFPB method, B-spline, linear interpolation, circular interpolation are prepared for trajectory planning. Fig. 5 depicts the scene of the teaching dialog box and the scene of the robot which automatically change its posture by modifying dialog box values.

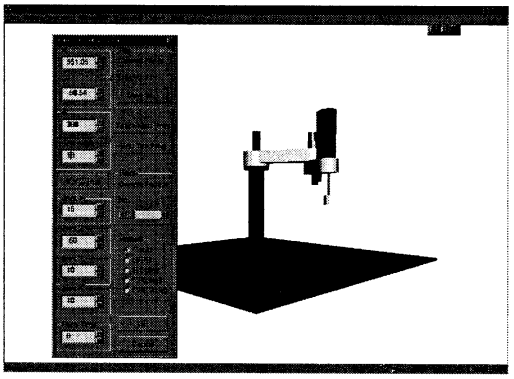


Fig. 5 Input Via Point Information in Teach Dialog Box

3.3.4 Dynamic simulation and performance evaluation and prediction

In Simulation menu, users can see 3D and 2D graphic views of the dynamic simulation in advance. This function can be done when trajectory planning sessions are completely finished. Users can evaluate the tracking performance of the selected control algorithms by checking graphic simulation. After this, users can decide which control algorithms are suitable for the given task. The provided control algorithms are PD control, computed torque method which is based on the dynamic analysis of real robot, and sliding mode control which is based on the variable structure system control. Users can select one in these three algorithms. Fig. 6 is the selection scene of a control algorithm in the three provided control algorithms, and Fig. 7 shows the scene of the robot in simulation which follows the given trajectory.

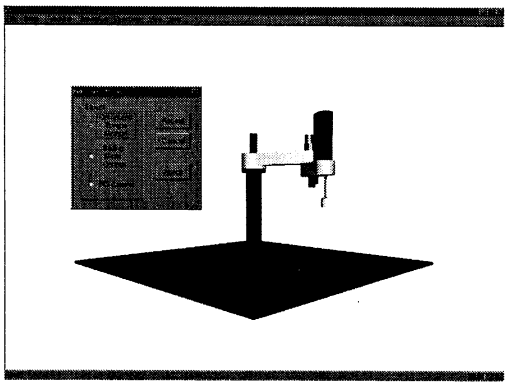


Fig. 6 Selection of Control Method

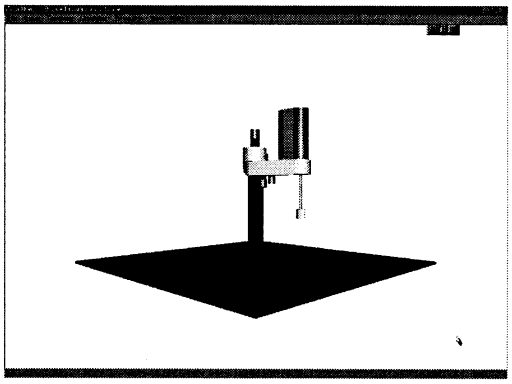


Fig. 7 Simulation in 3D Graphic Space

3.3.5 Other functions

In other menus, moving robot in desired posture, view point change, wire frame model view are also available. Fig. 8 is a combined scene of posture change and view point change. Especially, view point change can be freely done by the mouse movement. After wire frame options are selected, the function depicted in Fig. 9 is available.

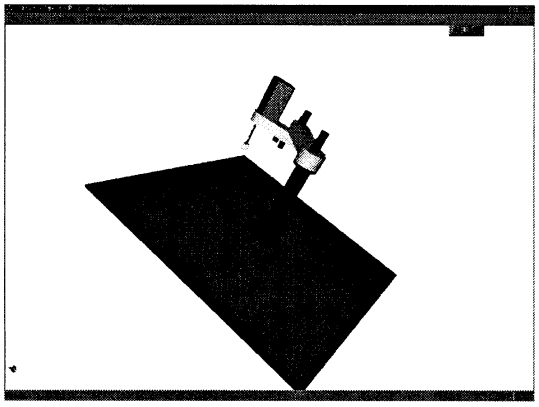


Fig. 8 Posture, View Point Change of SCARA Robot

4. Program Developing Environment

The developed OLP system is encoded by using Microsoft Visual C++ version 4.0 language and Silicon Graphics Inc.'s graphic library OpenGL. Visual C++ provides MFC and various Wizard(this function automatically encodes basic sources for user convenience), which enable users to make Windows application easily. Because of this, the developed program can be simply expanded or simply replanted on other programs. In addition, various visual editors are also conveniently used for making designed dialog box and menu bars[4]. The utilized OpenGL will be widely used for next industrial standard graphic library[5]. Because this library provides fast 3D graphic ability in Windows 95 environment, and has simple commands to use, it is very useful for program development.

5. Conclusion

In this paper, a 3D Graphic simulation tool for a 4 axes SCARA robot is developed. The tool is designed to operate in Windows 95 environment. Trajectory planning, kinematics, dynamics, and three control algorithms are incorporated in the developed program. GUI environment menu bar and dialog box are also developed for the system. Users can command the robot and evaluate all simulation situations using the developed tool by the monitor view.

Further studies are required on the connection between real robot motion and virtual space simulation.

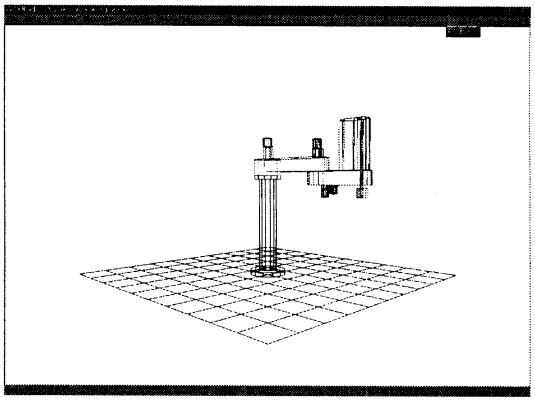


Fig. 9 Wire Frame Model

References

[1] R. Bolles, and B. Roth, *International Symposiums of Robotics Research,* MIT Press, Cambridge, MA, 1988.

[2] J. J. Craig, *Introduction to Robotics Mechanics and Control,* Second edition Addi son-Wesley, New York, 1989.

[3] G. Wittenberg, "Developments in Off-line Programming: An Overview," *Industrial Robot,* Vol. 22, No. 3, pp. 21-23, 1995.

[4] D. J. Kruglinski, *Inside Visual C++ 4,* Microsoft Press, 1996.

[5] R. Fosner, *OpenGL Programming for Windows 95 and Windows NT,* Addison -Wesley Developers Press, 1997.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Polynomial Fuzzy Neural Network for Modeling and Control

Sungshin Kim* and Man Hyung Lee**

*Department of Electrical Engineering
**Department of Control and Mechanical Engineering
& ERC/Net Shape & Die Manufacturing
Pusan National University, Pusan, Korea
sskim0@hyowon.pusan.ac.kr, mahlee@hyowon.pusan.ac.kr

Abstract— This paper introduces a novel neuro-fuzzy system based on the polynomial fuzzy neural network (PFNN) architecture. The PFNN consists of a set of if-then rules with appropriate membership functions (MFs) whose parameters are optimized via a hybrid genetic algorithm. A polynomial neural network is employed in the defuzzification scheme to improve output performance and to select appropriate rules. A performance criterion for model selection is defined to overcome the overfitting problem in the modeling procedure. For a performance assessment of the PFNN inference system, two well-known problems are employed to compare with other methods. The results of these comparisons show that the PFNN inference system outperforms the other methods and exhibits robustness characteristics.

Keywords– Polynomial neural network, Fuzzy Logic, Hybrid genetic algorithm, Dynamic Coding

I. Introduction

Fuzzy model identification that is based on fuzzy implications and reasoning is one of the most important aspects of fuzzy system theory because of its simple form as a tool and its power for representing highly nonlinear relations. Using qualitative expressions, graded numbering, and experimental data, the performance index or yield can be incorporated into a fuzzy model, a fuzzy objective function, or a decision making model [10], [14], [15]. The structure of a fuzzy model is the same as that of fuzzy control rules. In other words, a fuzzy model describes the features of the process using an *if-then* form that consists of the premise and consequent parts.

In this paper, we introduce a new neuro-fuzzy system, an effective optimization method through a genetic algorithm (GA) [3], a performance criterion for model selection, and a numerical example to illustrate the proposed modeling approach.

II. Polynomial Fuzzy Neural Network

The proposed neuro-fuzzy system is based on the polynomial fuzzy neural network (PFNN) architecture, as shown in Figure 1. The fuzzy inference engine based on the PFNN is combined with the "Type I" fuzzy system described in [5], through a polynomial neural network (PNN). A polynomial neural network [6] is employed in the defuzzification scheme to improve output performance and select rules. A PNN is a feed-forward network that computes a polynomial function of a set of parallel inputs in order to generate an output.

The PFNN consists of a set of if-then rules with appropriate MFs whose parameters are optimized via a hybrid genetic algorithm discussed in Section III.. For n inputs, x_i, with m_i MFs, $i = 1, \cdots, n$, and one output, $\hat{y}$, the total number of rules, r, is defined as: $r = \Pi_{i=1}^{n} m_i$. A typical rule with fuzzy if-then structure is expressed by:

$$Rule_{(i)}: \quad \text{If } x_1 \text{ is } A_1^{i_1} \text{ and } \cdots \text{ and } x_n \text{ is } A_n^{i_n} \quad \text{Then } y_i \text{ is } p_i \ , \tag{1}$$

where x_j, $j = 1, \cdots, n$, are non-fuzzy input variables, $A_k^{i_k}$, $1 < i_k < m_k$, are fuzzy variables related to the k-th input with i_k-th MF, and p_i, $i = 1, \cdots, r$, are constants in the consequent. The MFs for A_i^j, $\mu_{A_i^j}(x_i)$, for example, can be characterized by a Gaussian MF: $\mu_{A_i^j}(x_i; c_j, \sigma_j) = \exp\left\{-(x_i - c_j)^2/(2(\sigma_j)^2)\right\}$. Parameters $\{c_j, \sigma_j\}$ in this layer are referred to as premise parameters. The outputs of layer 4, R_i, are inferred as follows:

$$R_i = \frac{\prod_{k=1}^{n} \mu_{A_k^j}(x_k) \cdot p_i}{\sum_{j=1}^{r} \prod_{k=1}^{n} \mu_{A_k^j}(x_k)} \ . \tag{2}$$

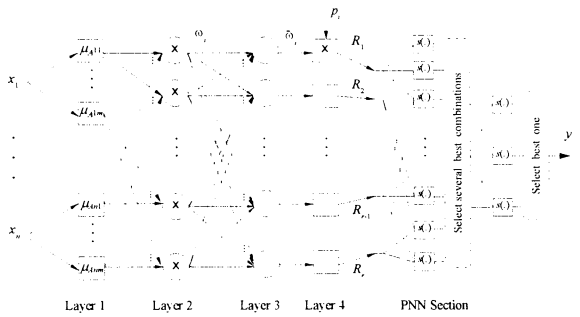


Figure 1. Polynomial fuzzy neural network.

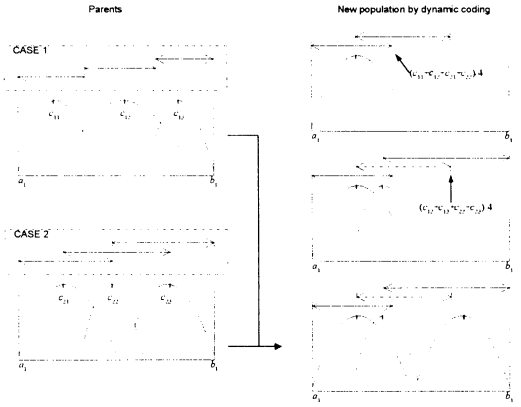


Figure 2. Dynamic coding method for fuzzy rule bases.

The parameters $\{p_i\}$ in this layer are called consequent parameters and are optimized by the method discussed in Section III.

In the PFNN model, the outputs in layer 4, R_i, are considered as inputs to the PNN. There are three main reasons why a PNN is used and why the R_i's are employed as inputs to the PNN: (1) After passing the R_i's through the PNN algorithm, the overall output, $\hat{y}$, is more accurate than the sum of the R_i's since the PNN, as a inductive learning algorithm, gives a better approximation than the sum of the inputs. (2) The output of layer 4 is the best input candidate since it is unbiased. (3) The PNN selects several R_i's from among all of the R_i's, thereby reducing the number of rules.

In the PNN section, a least-squares fit of the training data for the function to be learned is determined for all pairs of the input variables, R_i, using polynomials of up to second degree. The output of the bottom left module, for example, is a function of the inputs R_{r-1} and R_r:

$$s(R_{r-1}, R_r; a_i) = a_0 + a_1 R_{r-1} + a_2 R_r + a_3 R_{r-1} R_r + a_4 R_{r-1}^2 + a_5 R_r^2 \quad (3)$$

III. Genetic Algorithms for Fuzzy Systems

A. Hybrid Genetic Optimization

Genetic algorithms are a class of general purpose search strategies that strike a near optimal balance between exploration and exploitation of a parameter space. The major disadvantage of a genetic algorithm is the excessive number of runs of the design code required for convergence. The search space in a GA is discretized by its resolution. In the binary coding method, the bit length L_i and the corresponding resolution R_i are related by $R_i = (UB_i - LB_i)/(2^{L_i} - 1)$, where UB_i and LB_i are the upper and lower bounds of the parameter x_i, respectively. By adding k bits to the parameters, the resolution is improved by approximately 2^k times. On the other hand, the search space is dramatically increased to $(2^k)^n$ times. To overcome this problem, a hybrid genetic optimization method that combines a GA with Nelder and Mead's simplex method is introduced. First, a GA is used to search the coarse search space and to find the basin of attraction of the global solution. Second, the result of the GA is passed to a simplex method as an initial condition.

B. Dynamic Coding Method

The range of p_i in (1) and σ_i in MFs will be determined from the GA, but the range of the center value, c_i, in the MF cannot be fixed. A fixed range for c_i will degrade the performance of the fuzzy system, because it reduces the search space or may permit the center values of the MFs to be swapped. These are shown in Figure 2 as Case 1 and Case 2, respectively.

In the proposed dynamic coding method, the range of c_i in each generation is dynamically varied from parents to children. Figure 2 shows the dynamic coding method graphically. This method will maximize the search space and will not permit overlap behavior between MFs.

IV. Performance Criterion

A most common and difficult problem in the empirical modeling arena is the question of when to stop searching the free parameters or adding terms to a model. The proposed performance criterion (PC) for model selection is based on the GMDH so that the error is minimized and, at the same time, overfitting of the empirical data set is prevented. As a proper criterion for the verification of a model, the observed data are

Model Name	Number of Inputs	Number of Rules	Model Error
Box *et al.*[1] ,[2] [1]	6	–	0.202
Tong's model [13]	2	19	0.469
		25	0.776
Pedrycz's model [8]	2	49	0.478
		81	0.320
Xu's model [17]	2	25	0.328
TSK model [11]	2	2	0.359
Linear model[1] [12]	5	–	0.193
Sugeno *et al.* [12]	3	6	0.190
Kupper's model [7]	2	25	0.166
Saleem *et al.* [9]	4	–	0.417[3]
	3	–	0.403
Wang *et al.* [16]	2	5	0.158
Our model	2	11	**0.108**

[1]Non fuzzy model. [2]ARMA model. [3]NN (4-4-1) model.

Table 1. Comparison of PFNN model with other models

divided into two sets: N_A for training and N_B for testing purposes. The performance criterion is defined as:

$$\begin{aligned} e_1^2 &= \sum_{i=1}^{n_A} \left(y_i^A - f_A\left(x_i^A\right)\right)^2 / n_A \ , \\ e_2^2 &= \sum_{i=1}^{n_B} \left(y_i^B - f_A\left(x_i^B\right)\right)^2 / n_B \ , \\ PC &= e_1^2 + e_2^2 + \eta\left(e_1^2 - e_2^2\right)^2 \ , \end{aligned} \tag{4}$$

where n_A is the number of data points in the data set N_A and y_i^A is the real output of the data set N_A.

The objective is to minimize the PC so that the best model for N_A and N_B may be found. The final model is not overfitting the training data and it is optimal on the basis of the data sets N_A and N_B without bias.

V. Performance Test and Evaluation

A. Box and Jenkins's Gas Furnace

This section presents an example of fuzzy modeling using the gas furnace data presented in Box and Jenkins [1]. The collected data includes 296 successive input-output pairs of observations, $(u_t,\ y_t)$. The performance index (PI) used for comparison with other identification algorithms is formulated as: $PI = \frac{1}{N}\sum_{k=1}^{N}(y(k) - \widehat{y}(k))^2$.

The fuzzy model is assumed to take the form $y(t) = F(y(t-1), u(t-4))$. The initial values used in the GA are: number of populations=100, number of generations=300, crossover rate=0.75, and mutation rate =0.001. The performance of the proposed model is compared with other fuzzy models in Table 1.

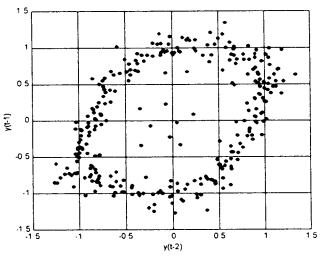


Figure 3. The noisy iterated dynamics of the time series from an initial position at the origin.

B. Nonlinear Time Series (Limit Cycle) Modeling

A second benchmark problem to test the performance of the proposed scheme is the time series function described by the following second-order nonlinear difference equation:

$$\begin{aligned} y(t) = &\left(0.8 - 0.5\exp\left(-y^2\left(t-1\right)\right)\right) y(t-1) - \\ &\left(0.3 + 0.9\exp\left(-y^2\left(t-1\right)\right)\right) y(t-2) + \\ &\sin\left(\pi y(t-1)\right) + \eta(t) \ , \end{aligned} \tag{5}$$

where $\eta(t)$ denotes the additive noise at time t. When $\eta(t) = 0$, this difference equation has an unstable equilibrium at the origin and a globally attracting limit cycle. Different modeling methods, such as CMAC and B-splines in [4], and RBF and MLP in [2], have been applied to the same problem.

In this problem, $\eta(t)$ has been assumed as a zero-mean Gaussian white noise sequence with variance 0.01. A set of noisy training samples are collected from (5) at zero initial condition with 300 iterations and shown in Figure 3.

After training with the proposed method, the identified PFNN model is evaluated using two different tests. In the first one, the normalized output error autocorrelation function, $\phi(k)$, is used to evaluate the approximation ability over the training set, and is computed as: $\phi(k) = \sum_{t=1+k}^{N}\{\varepsilon(t)\varepsilon(t-k)\} / \sum_{t=1}^{N}\{\varepsilon(t)\}^2$, where k is the time lag, N is the training set size, and $\varepsilon(t)$ is the output error at sample time t over the training set. When the identified model reproduces the underlying function exactly, $\phi(k)$ becomes an impulse function at $k = 0$; i.e., $\phi(k)$ is equal to one at $k = 0$, and zero otherwise. The 95% confidence limits are $\pm 1.96/\sqrt{N}$. The autocorrelation plot for the PFNN model is shown in Figure 4, and the values lie within the 80.6% confidence band rather than the 95%. Thus, there has been an approximate improvement of 14.4%, whereas other methods are barely inside or outside the 95% confidence band.

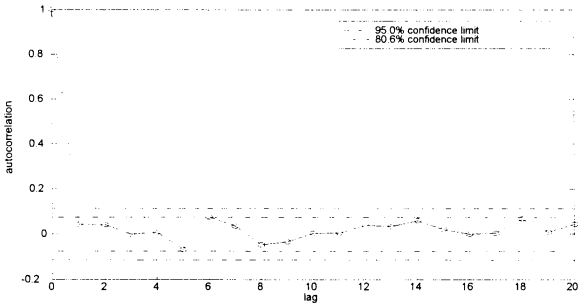


Figure 4. The autocorrelation of the prediction errors after the 20th training cycle. The 95% confidence band is shown by the dashed lines.

The second test for assessing the model's performance is to iterate the identified model from an initial condition $\mathbf{x}(1) = (y(0), y(-1)) = (0.1, 0.1)$, and then plot the resulting dynamic behavior. The phase plot and the time series surface plot are shown in Figure 5. The basic shape of the limit cycle is a good approximation to the true shape, and a five armed interior spiral can be seen clearly.

VI. Summary and Conclusions

This paper proposes a new neuro-fuzzy system, an effective optimization method through a hybrid genetic algorithm and a performance criterion for model selection. The neuro-fuzzy system, based on the polynomial fuzzy neural network architecture, improves the defuzzification scheme.

The illustrated examples demonstrate the effectiveness of the proposed modeling approach. A PFNN modeling and hybrid genetic optimization method can be employed to provide optimum setpoints for a control activity.

Acknowledgment

This work was supported by ERC/Net Shape & Die Manufacturing, and the Research Fund of the POSCO Chair Professor at Pusan National University.

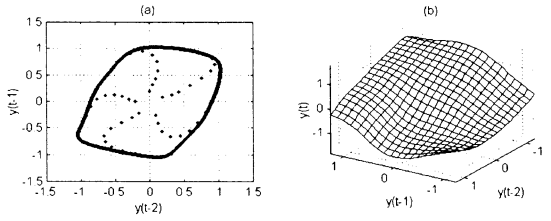


Figure 5. (a) The iterated dynamical behavior of the PFNN from the initial condition $x(1) = (0.1, 0.1)^T$. (b) The PFNN approximation of the true time series surface.

References

[1] G.E.P. Box and G.M. Jenkins, *Time Series Analysis, Forecasting and Control*, San Francisco, Holden Day, 1976.

[2] S. Chen and S. A. Billings, "Neural networks for nonlinear dynamic system modelling and identification," in *Advances intelligent Control* (C. J. Harris, ed.), ch. 4, Taylor and Francis, 1994.

[3] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

[4] C. J. Harris, *Neurofuzzy Adaptive Modelling and Control*, ch. 6 and 8. Prentice Hall, 1994.

[5] S. Horikawa, T.Furuhashi, and Y. Uchikawa, "On fuzzy modeling using fuzzy neural networks with the backpropagation algorithm," *IEEE Trans. Neural Networks*, vol. 3, no. 5, pp. 801-806, 1992.

[6] A. G. Ivakhnenko, "Polynomial theory of complex systems," *IEEE Trans. Syst. Man and Cybern.*, vol. SMC-12, pp. 364-378, 1971.

[7] K. Kupper, "Self learning fuzzy models using stochastic approximation," *Proc. of the 3rd IEEE Conf. Control Applications*, vol. 3, pp. 1723-1728, 1994.

[8] W. Pedrycz, "An identification algorithm in fuzzy relational systems," *Fuzzy Sets and Systems*, vol. 13, pp. 153-167, 1984.

[9] R. M. Saleem and B. E. Postlethwaite, "A comparison of neural networks and fuzzy relational systems in dynamic modelling," *Int. Conf. Control '94*, vol. 2, p.1448-1452, 1994.

[10] S. Kim, A. Kumar, J.L. Dorrity, and G. Vachtsevanos, "Fuzzy modeling, control and optimization of textile processes," *Proc. of the 1994 1st Int. Joint Conf. of NAFIPS/IFIS/NASA*, San Antonio, TX, pp. 32-38, Dec. 1994.

[11] M. Sugeno and K. Tanaka, "Successive identification of a fuzzy model and its application to prediction of a complex system," *Fuzzy Sets and Systems*, vol. 42, pp. 315-334, 1991.

[12] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *IEEE Trans. Fuzzy Systems*, vol. 1, no. 1, pp. 7-31, 1993.

[13] R. M. Tong, "The evaluation of fuzzy models derived from experimental data ," *Fuzzy Sets and Systems*, vol. 4, pp. 1-12, 1980.

[14] G. J. Vachtsevanos, S. S. Kim, J. R. Echauz, and V. K. Ramani, "Neuro-Fuzzy Approaches to Decision Making: A Comparative Study with an Application to Check Authorization," *Journal of Intelligent and Fuzzy Systems*, vol. 6, pp. 259-278, 1998.

[15] G. Vachtsevanos, J.L. Dorrity, A. Kumar, and S. Kim, "Advanced application of statistical and fuzzy control to textile processes," *IEEE Trans. on Industry Applications*, vol. 30, no. 3, pp. 510-516, May-June 1994.

[16] L. Wang and R. Langari, "Complex systems modeling via fuzzy logic," *IEEE Trans. Syst. Man Cybern.*, vol. 26, no. 1, pp. 100-106, Feb. 1996.

[17] C.W. Xu, "Fuzzy systems identification," *IEE Proceedings*, vol. 136, Part. D, no. 4, 1989.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Development of Horseback Riding Therapy Simulator with VR Technology

Osamu Sekine, Youichi Shinomiya
Ryoji Nakajima
Advanced Technology Research Laboratory
Matsushita Electric works, Ltd.
1048, Kadoma, Osaka, 571-8686, Japan

Tetsuhiko Kimura

Department of Health Services Administration
Nippon Medical School
1-1-5, Sendagi, Bunkyo, Tokyo, 113-8602, Japan

Abstract

We have developed a system that reproduced the effect of horseback riding therapy, which device may be enjoyed by healthy senior citizen, and has been specifically designed to rejuvenate both the mind and the body. We have also developed a Virtual Reality section that is effective in creating images and sound that complement both each other and correspond to the movements produced by the drive system. The controlling action mechanism of the Virtual Reality Operation section allows interactive starting, speeding up, slowing down, turning and stopping through commands given through the reins or by leg movements as auxiliary functions.

Keywords: Horsebback Riding Therapy, Virtual Reality, Stewart Platform

1 Introduction

It has been said since the time of bygone days that horseback riding gives both physical and psychological benefits in the rider, and horseback riding has often been used to help treat disabled persons and for its curative effects in general, especially in Northern Europe. It is getting more and more attention in Japan as well as horseback riders gain a magnificent "sensation of release" by riding a horse outdoors and through the elevation of the level of eyesight. In addition, the "dialogue with a living animal" that takes place every time a rider swings up onto a saddle and the nurturing involved in caring for an animal bring psychological benefits to the rider. The activities of maintaining or recovering one's balance, shifting one's posture and sifting one's weight in response to a horse's movements are part of the process of giving physical simulation and are singled out as some of the positive aspects of the exercise.[1] However there are few examples of reports that state that these balance maintaining functions, posture shifting function and possibilities for foot training through up-and-down movements give the same benefits to everyone.[2] We have developed system that reproduced the effect of horse back riding therapy. This training device may be enjoyed by healthy senior citizens, and it has been specifically designed to rejuvenate both the mind and the body. By conducting horseback riding cures with a machine, it is possible to accurately control the distribution of loads, and the effects of bearing these loads on the human body can be evaluated.

Since long ago, there have been a number of reports of horseback riding simulations, but these simulators were not developed specifically in order to include the additional function in application to medical research. The movements of the saddle are reproduced at predetermined points.[3][4][5] One has attempted to make the simulator simulate the walking gait with a multi-legged walking movement.[6][7] And the movements of the horse are brought to life through the use of computer graphics.[8]

We have constructed a simulator that accurately reproduces the movements of saddle at predetermined points in order to mimic actual horseback riding and to guarantee the safty of the rider.

2 Analysis of Actual Riding Data

Before designing the actual simulator, we collected and analyzed data on the movements of the saddle during actual horseback riding. Only then did we put together our design for the physical simulator. We collected data on the movements by making a mark on a horse's saddle, filming the horse's and saddle's movements using high-speed photography, and then using a special method to make the 3-dimensional data. The locations of the mark are shown in Fig. 1, and the layout of the overall design is shown in Fig. 2.

Since it is said that the walking gait of horse is the most effective at producing the curative effects of

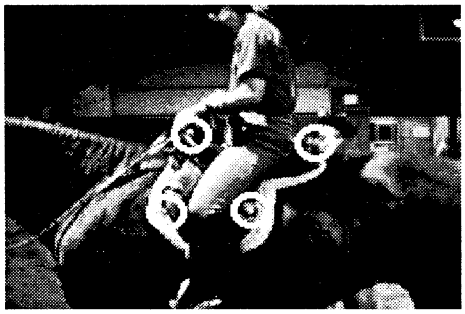


Figure 1: The location of the mark

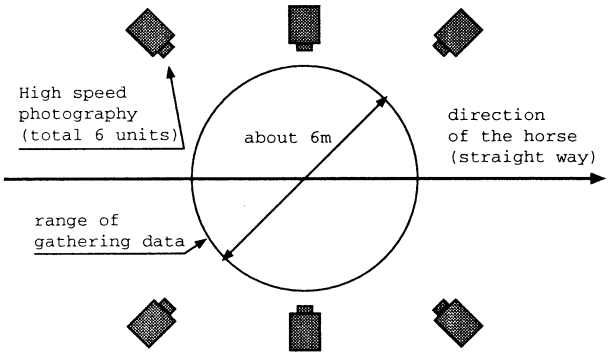


Figure 2: The layout of measuring

horseback riding, data was analyzed with the goal of duplicating the normal walking motion. Also, for the purpose of providing enjoyment for the rider, data was analyzed to produce the trot and canter gaits as well.

3 System Configuration

A diagram of the overall configuration of the system is provided in Fig 3. The configuration of the system can be divided up into four large sections: the drive section, the control section, the virtual reality section, and the operating section. Two computers are used to handle the CPU functions for ensuring that the drive section operates safety and to generate the virtual imaging. Explanation for all the section are given below.

3.1 Drive Section and Control Section

We wanted a small, simple, light-weight structure for the simulator drive section, which has the responsibility of giving life to the data derived for all the

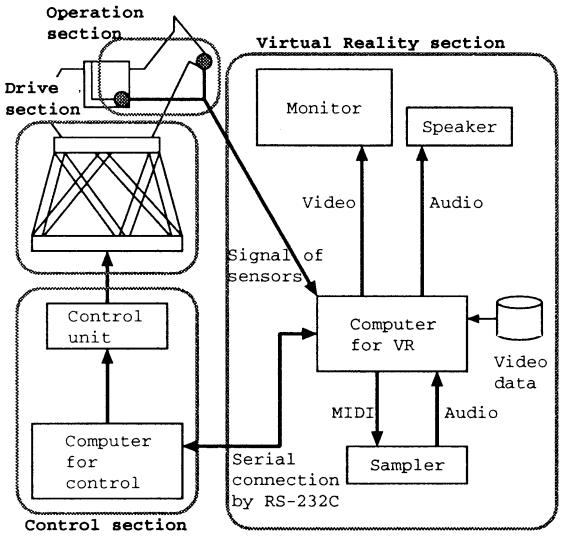


Figure 3: A diagram of the system configuration

walkig postures. We thought that a Stewart platform-shaped parallel mechanism would be appropriate for fulfilling these requirements. The appearance of total system is shown in Fig 4

In a typical direct drive actuator, oil pressure, air pressure, and motor driving are employed, but in this case, AC servo motor driving was selected, because it allows precise positioning. The capacity required for the AC servo motor was calculated from the specifications for the drive mechanism for reproducing the walking gait (link producing power, acceleration, and speed), and the design values for the transmission mechanism (rate of speed reduction, lead, weight of parts, and load). The capacity turned out to be 750W. Resin nuts were employed so that noise would be muffled. The current specifications were developed with an eye towards the medical benefits of horseback riding, so the system was designed in such a way that it will not produce violent motions. This also ensures that the elderly can use the machine without worry. The amplitudes and acceleration have been suppressed in both the trot and canter gaits.

The inverse kinematics calculation for obtaining the link length for realizing these factors when the location and position of the end effector are given are public knowledge.[9] Based on these calculation methods, the amount of control required for the six axes is calculated for every units of sampling time. Next, motor control is exerted to a degree sufficient to realize these control amounts.

In the parallel mechanism used in this system, the

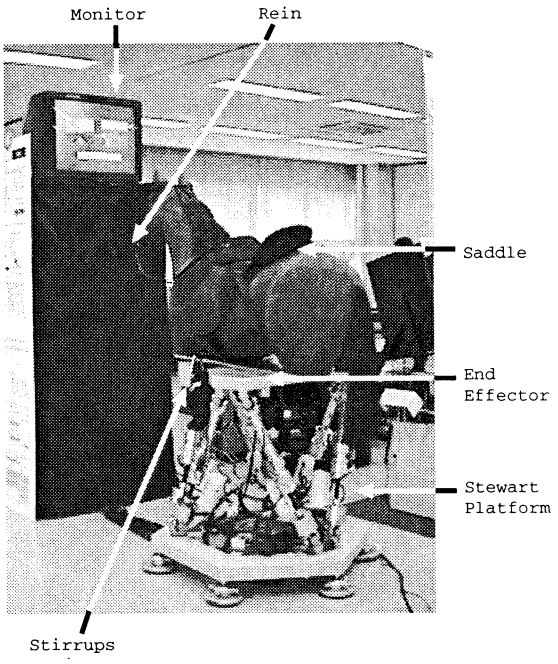


Figure 4: The appearance of total system

link length is increased or reduced by the rotation of the link axis, and it was necessary to develop an algorithm for correcting the increase or decrease of the link length because of deterioration in the precision of the positioning of the end effector. Due to the nature of the purposes for which this system is designed, which matches the value ordered by the end effector, be calculated in real time. For this reason, we developed a simple correction algorithm that would decrease the amount of calculation time. As a result, calculations can be made to within 1 ms, and a positioning precision of 0.3 mm has been obtained in the horizontal direction.

4 Virtual Reality Section

In normal horseback riding, the horse is controlled according to the wishes of the rider through rein controls, leg controls, and shifting of the hips to produce starting, increasing and decreasing in speed, turning, stopping, different kinds of gaits, and so on. By simultaneously displaying the sights and sound that normally accompany these actions that influence the horse, it is possible to train the rider and give him or her the surprisingly real sensation of actual riding a horse. This means that the simulator can help train the rider while making him or her enjoy the images and sound through the virtual reality system.

the following control behaviors are possible:

1. The horse moves as desired.

   Changing speed The speed is changed by changing the acceleration and amplitude.

   Changing Direction The direction is changed by increasing or decreasing of possible yaw and pitch.

   Changing the walking style The walking style is changed by changing the data loaded into the system.

2. Visible scenery

   The scenery changes Digitalized, subjective images of movement taken from real pictures are played back to match the scenes. When the speed of the drive section is changed, the imaging speed is changed.

   A wide range of vision Because a fish-eye lens was used for taking pictures, the angle of vision is quite large.

3. Auditory effects

   The sound of the hoofs The sound of hoofs change to match changes in speed and movements of the drive section.

   The sound of the environment When the frame is actually played back, the appropriate sound of the environment are played.

4. Makes the movements displayed in the image actually occur.

   The physical effects of a road The sensation of riding on a hill or curve is produced through the operation of the drive section. On an inclining road, the value of the yaw is changed.

5. Designed to always be exciting

   The progress of the road changes The system displays images that correspond to movement in the direction the rider wishes to ride by performing playback in either the A direction or the B direction, that is, the one which corresponds to movement in the direction to which the reins are pulled.

4.1 Virtual Reality Operation Section

As for the control movements, a system has been developed for both starting and increasing with leg controls, and turning and decreasing speeds with the rein controls.

Rein control movements are used for turning and slowing down. The "horse" holds an aluminum plate in its mouth, and sensors are attached to both ends of this plate. The reins and sensors are joined together so that when the reins are pulled forcefully, a signal is given by the sensors. When the sensors emit such a signal in the forward motion direction selection scenario, orders are given for the horse to advance in the direction in which the reins are pulled. In other scenarios, the speed will decrease by one notch when the reins are pulled.

Again, leg control movements are used for the starting and speeding up as an auxiliary function. A sensor which senses small shocks and turns the switch to ON is attached beneath the saddle, and when the rider provides shocks with his or her legs, a signal is given by the sensor. When the sensor emits such a signal in the stop scenario, the system acts so that starting occurs and one notch of speed is produced under the moving scenario.

If this system is to make good on its promise of providing rejuvenation for both the mind and body, then it is extremely important that an appropriate riding atmosphere be created along with the reproduction of the actual movements of a horse. Paying thorough attention to this aspect of the system, we have created a horse's body that comes close to the real thing. It was determined that the horse's body should be about the size of a pony, since this size has been judged to give the most health benefits from horseback riding. The main body is constructed of FRP, and suede has been stretched over the FRP to give a realistic "horse-like" feeling to the touch. A saddle, reins, and stirrups have also been provided.

5 Conclusions

People weighting up to 100 kilograms can ride the horseback riding simulator we have just developed. They will experience the sensation of riding a real horse as it walk, which is key to experiencing the health benefits of horseback riding. The response speed and positioning precision of the drive section were good enough for the system's performance to produce the walking gait. In addition, the noise produced by the drive system has been muffled to the degree that is entirely satisfactory.

We have also developed a Virtual Reality section that is effective in creating images and sound that complement both each other and correspond to the movements produced by the drive system. The controlling action mechanism of the Virtual Reality Operation section allows interactive starting, speeding up, slowing down, turning, and stopping through commands given through the rein or by leg movement as auxiliary functions, and the construction of the horse's body also provides the illusion of riding a real horse.

We will continue to do research on the influence of the movements involved in horseback riding on people and carry out measurements of the muscular effects produced in the human body using electrode sensors from this point on. We also have plans to evaluate the influence of movements transmitted by sources besides horse.

References

[1] Emiko Ohta (1997), The Curative Effects of Horseback Riding (in Japanese), *Stock Breeding Research*, Vol. 51, No. 1, pp. 148-154

[2] Tetsuhiko Kimura (1997), The Development of an Intelligent Simulator of Horses (in Japanese), *Medical Treatment*, Vol. 40, No. 8, pp. 749-755

[3] M. Yamaguchi et. al. (1992), Development of a Horseback Riding Simulator, *Advanced Robotics*, Vol. 6, No. 4, pp. 517-528

[4] Jouffroy J. J. (1991), L'analyse et la Restitution des Sensations par Simulation en Equitation: Programme Persival, *Science & Sports*, No. 6, pp. 129-131

[5] Y Amirata et. al. (1996), The Design and Control of a New Six Dof Parallel Robot: Applications in Equestrian Gait Simulation, *Mechatronics*, Vol. 6, No. 2, pp. 227-239

[6] E Koizumi (1996), Dynamic Trot Control in a Quadruped Robot (in Japanese), *The Japan Machinery Society, Robotics/Mechatronics Lecture Meeting 96, Lecture Collection*, Vol A, No. 96-2, pp. 301-304

[7] E Koizumi (1994), Gallop Gait by a Quadruped Locomotion Robot Scamper (in Japanese), *The Japan Machinery Society, The 7th Intelligent Moving Robot Symposium Lecture Collection*, No. 940-252, pp. 32-36

[8] P. Koenig et. al. (1993), The Generation and Control of Gait Patterns in a Simulated Horse, *Proc. IEEE International Conf., Robots and Automatics*, pp. 359-366

[9] Japan Robot Society (1993), Parallel Mechanism Research Special Committee Report (in Japanese), pp. 85-98

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Novel Application of Face Image Processing to Wearable Computer - Augmentation of Human Memory for Faces and Names -

Hiroshi Mizoguchi Takaomi Shigehara Yoshiyasu Goto Taketoshi Mishima
Faculty of Engineering
Saitama University
255 Shimo Okubo, Urawa 338-8570, Japan

Abstract

This paper proposes a novel application of face image recognition to a wearable computing system that augments human memory for face. The proposed system uses small sized CCD camera attached to a see-thorough type head mounted display(HMD). It identifies input face image and superimposes his or her name on the HMD screen. Not only the name, date last met and other information related to the customer are also displayed. For a user of this system, it works as if he or she were accompanied by an invisible assistant having excellent memory for faces, names, dates and so forth.

1 Introduction

The rapid advance of computers, especially personal computers, has remarkably changed and increased efficiency of our everyday work in recent years. However, situation and field where benefits of such advances are possible to apply to are still limited to desk top in ordinary offices and other in door scene. Quite number of people cannot obtain such benefits while they are walking, moving, talking and meeting others. In other words, novel application fields of computers still remain in our dairy working scene. Thus novel device and technology to expand the fields are expected and researched[1]-[8].

A typical example of the remaining fields is working situation of a salesperson who must often visit customers. For the salesperson fighting severe business competition, it is very crucial to recall customer's face, name, date last met and other various information related to the customer, especially when he or she meets the customer face-to-face. If a computer were embedded in eyeglasses and could recognize and remember the faces, it would be very convenient and helpful for the salesperson.

Therefore this paper proposes a novel application of face image recognition to a wearable computing system that augments human memory for face. The proposed system uses small sized CCD camera mounted on a see-thorough type head mounted display(HMD) and identifies input customer's face image and superimposes his or her name on the HMD screen. Although the current CCD camera and HMD are still so large comparing to ordinary eyeglasses, it can be expected that these components will become so small that can be embedded in the eyeglasses. And the state-of-the-art embedded computer is also too large both in size and power consumption, however it is not impossible to imagine such small future computer that can be embedded in temple or rim of eyeglasses.

Based upon recent progress in face recognition technology[9]-[15], increase of computing power, advancement of CCD and HMD, and demand for more sophisticated human interface, it is possible and significant to implement a functional prototype of the proposed face recognition system utilizing the not small sized current components. This implementation will prove feasibility of the proposed idea and contribute to examine required functions. The authors are currently implementing the functional prototype.

In the following, section 2 describes principle of the proposed system. In section 3, software issues of the proposed system are discussed. Section 4 describes currently implementing functional prototype. Section 5 is conclusion

2 Overview of the Proposed System

Fig. 1 shows a block diagram of the proposed system. This figure illustrates both principle of the proposed system and typical situation where the system is used. The system consists of face recognition part, retrieval part, and relational data base. As the input and output devices, CCD cameras and see-through type HMD(head mount display) are utilized. The face recognition part detects, tracks and recognizes human face continuously. To detect and track face area, skin color extraction method is utilized. If the recognition succeeds, its result is sent to the retrieval part. This part seeks for related information about the recognized person within the relational data

base. The retrieved information, such as name of the person, last meeting date, his/her profile, and so forth are superimposed on the HMD screen.

Ordinarily the user of the system is possible to see outside scene in real world via the see-though function of the HMD. And he/she can obtain annotation to the real scene by the superimposing information. This function is expected to apply not only face recognition but also navigation, instruction, maintenance, assembly, and so forth

3 Face Processing

The face recognition part has there functions, face detection, tracking and recognition.
The face detection is realized by skin color extraction. A method used in the skin color extraction part is based upon YIQ color representation instead of commonly used RGB one, because I and Q components of the YIQ are independent of brightness and free from brightness change. Original idea of the method is found in Mori et al.[16].

Fig. 2 shows IQ plane of the YIQ representation. Skin color is possible to be defined as some continuous region in the plane. Thus skin color area of the input image can be extracted by filtering each pixel of the input image based upon whether the corresponding point in the IQ plane is within the region or not. Fig. 3 shows a result of a preliminary experiment. Similar extracted areas can be obtained while brightness level of input images are so different as shown in this figure.

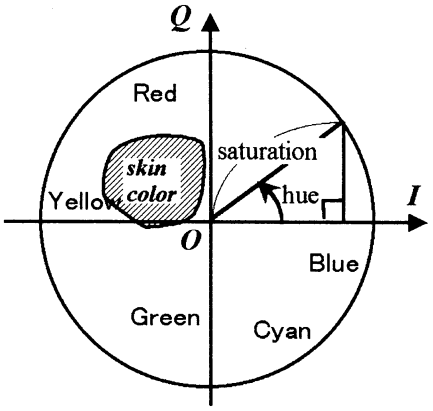
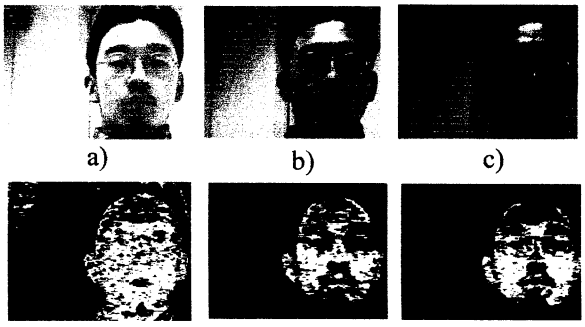


Fig. 2 IQ plane of YIQ color representation



a)-c): Input images of different brightness
d)-f): Skin color areas extracted from a)-c)

Fig. 3 Skin color extraction example

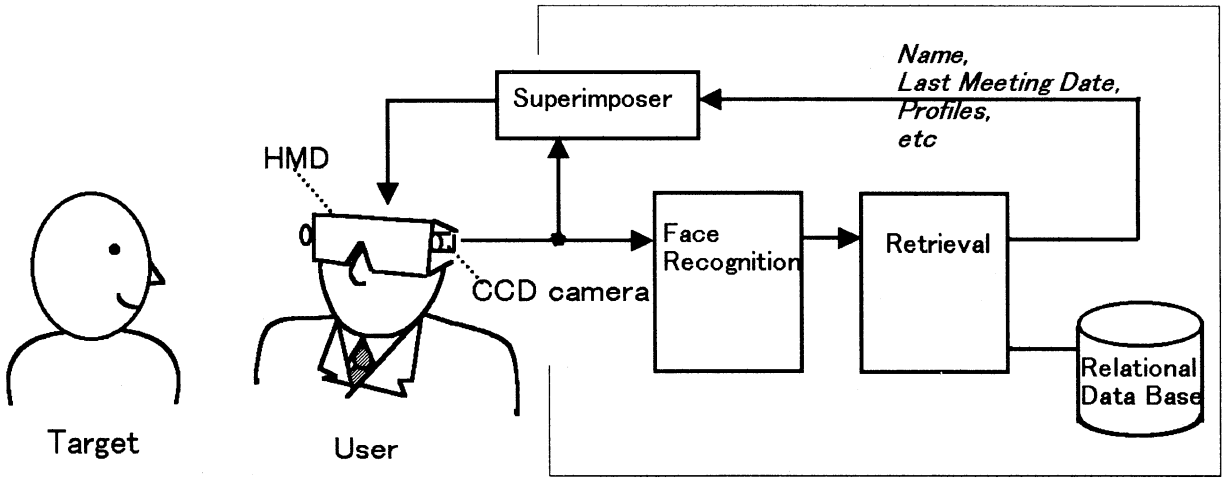


Fig. 1 Blockdiagram of proposed system

The extracted skin color area is a candidate of a face image. Within the extracted area, pattern matching process is performed to find whether the candidate is matched to previously registered face images or not. The matching process is based upon cross correlation. To calculate the cross correlation a dedicated hardware is utilized. The tracking function is also based upon cross correlation by the hardware. The function utilizes the correlation of previous and current frame images at each video frame.

The dedicated hardware to calculate to cross correlation of images is originally developed by Inoue et al. of Univ. of Tokyo[17][18][19] and commercialized by Fujitsu [20][21][22]. The authors utilize Fujitsu's product. It enables to calculate cross correlation of 8 by 8 template within search space at about 500 times during one video frame (1/30 sec). Since the calculation of cross correlation is performed at far beyond video rate by the hardware, the can be done in real time.

4 Implementation of Functional Prototype

To confirm the proposed idea, the authors are now implementing a functional prototype of the proposed system, utilizing commercially available components. As for the input device we use 1/4 in. color CCD camera board made by TOSHIBA. Sizes of the camera board including lens is 30mm x 30mm x 20mm. Fig.. 4 shows outlook of the CCD camera board.

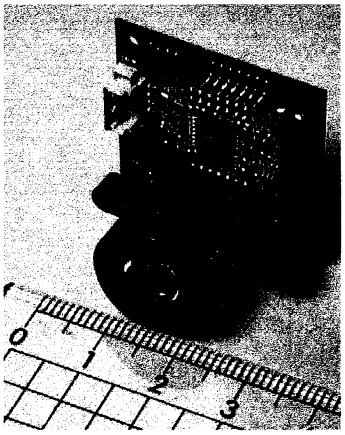


Fig. 4 Small sized CCD camera

As described above, we choose a see-through type HMD as output device. Fig. 5 shows outlook of the HMD equipped with the above mentioned color CCD cameras. The HMD is a consumer product manufactured by SONY. Its interface is NTSC, not VGA nor SVGA. Although the NTSC interface limits the displayed image to low resolution, we choose the product because of its easy availability, light weight, and low cost. A VGA-to-NTSC adapter is utilized to convert VGA signal from the computer.

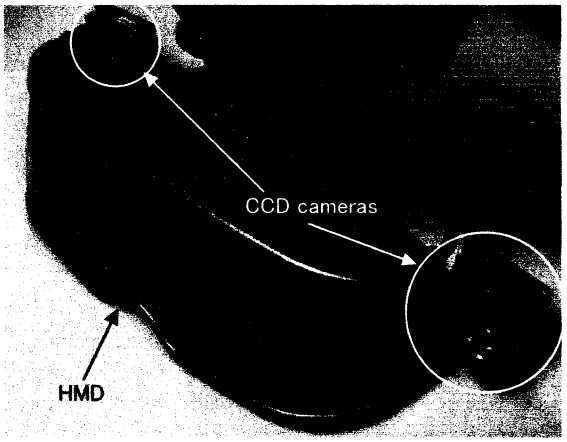


Fig. 5 HMD with CCD cameras

Processing software described in the previous section are executed on a PC compatible small sized embedded board computer for industry use. There are two reasons to choose the PC compatible. One is wide range of hardware from small sized board to large sized rack type. And, as for the PC compatible, there has been rapid and continuous growth of hardware performance. Even though commercially available board sized PC currently does not have so much high performance, we can expect the rapid and continuous growth and to obtain sufficient performance in the near future.

The other reason is wide range of software selection not only for applications but also operating systems(OS) and middle ware, such as data base management systems(DBMS). As for the operating system, Linux OS is utilized because it is light weighted and free but true preemptive multitasking OS. Since source code of Linux OS is freely available, it is relatively easy to modify and extend to introduce experimental functions. Thus Linux OS is more suitable than other commercial operating systems for this kind of research purpose experimental system. For Linux OS there are also various practical applications and number of such kind applications has increased recently. Moreover there are practically usable and high quality middle ware, such as web server and DBMS. As for the DBMS, PostgressSQL is under consideration and evaluation to use.

All hardware components described above, CCD cameras, HMD and PC compatible board computer, are powered by battery for portable use. In this prototype system, Lithium-ion battery for camcorder is utilized as a power source because of its high cost performance, energy per weight ratio and availability.

5 Conclusion

This paper proposes a novel application of face image processing to a wearable computing system. The proposed system utilizes see-through type head mounted display equipped with small sized CCD cameras as input and output devices. The system recognizes input face image and superimposes information related to the face. The proposed system is useful and effective for salesperson because it works as if it were an invisible assistant having good memory for faces and names of customers.

The proposed system is practically useful for salesperson as mentioned above, but is not limited to such application. The system can be viewed as an augmented reality(AR) device that superimposes annotations on real world image. It is a good tool to study effectiveness of the AR technology. Not only customer's name, the system is possible to unconsciously accumulate records of other information, such as, when, where and what the salesperson has seen, to whom and how long he has met, and so on. This accumulation is unexpectedly useful when the records are recalled and analyzed afterwards. Thus the proposed system is also effective tool for novel computer science research topics on wearable computer and affective computing.

At the time of writing this paper, the authors are implementing a functional prototype of the proposed system, using currently commercially available components. In this paper, implementation issues of the prototype are also described. Integrating elemental technologies, completing the prototype, conducting experiments and evaluating the prototype are future works.

References

[1] P. Maes, "Agents that reduce work and information overload", CACM, Vol.37, No.7, pp.31-40, 1994.

[2] M. C. Torrance, "Advances in Human-Computer Interaction: The Intelligent Room", Working Notes of the CHI95 Research Symposium, 1995.

[3] A. Pentland, "Smart Rooms", Scientific American, pp.54-62, 1996.

[4] B. J. Rhodes and T. Starner, "Remembrance agent: A continuously running automated information retrieval system", Proc. of First Int'l Conf. on the Practical Application of Intelligent Agents and Multi Agent Technology, pp. 487-495, 1996.

[5] R. W. Picard, Affective Computing, MIT Press, 1997.

[6] S. Mann, "Wearable computing: A first step toward personal imaging", Computer, pp.25-31, 1997.

[7] T. Starner, J. Weaver, and A. Pentland, "A wearable computer based American sign language recognizer", Proc. of the First ISWC, 1997.

[8] R. W. Picard and J. Healey, "Affective wearables", Proc. of the First ISWC, 1997.

[9] M. Turk and A. Pentland, "Face Recognition Using Eigenfaces", Proc. of CVPR'91, pp.586-591, 1991.

[10]T. Kurita, N. Otsu, and T. Sato, "A Face Recognition Method using Higher Order Local Autocorrelation and Multivariate Analysis", Proc. of ICPR'92, pp. 213-216, 1992.

[11]I. Craw, N. Costen, T. Kato, G. Robertson, and S. Akamatsu, "Automatic Face Recognition: Combining Configuration and Texture", Proc. of FG'95, pp. 53-58, 1995.

[12]T. Sakaguchi and S. Morishima, "Face feature extraction from spatial frequency for dynamic expression recognition", Proc. of ICPR'96, pp. 451-455, 1996.

[13]T. Kurita, K. Hotta and T. Mishima, "Scale and Rotation Invariant Recognition Method using Higher-Order Local Autocorrelation Features of Log-Polar Image, Proc. of ACCV'98, 1998.

[14]S. A. Rizvi, P. J. Phillips and H. Moon, "The FERET Verification Testing Protocol for Face Recognition Algorithms", Proc. of FG'98, pp.260-265, 1998.

[15]G. J. Edwards, C. J. Taylor and T. F. Cootes, "Learning to Identify and Track Faces in Image Sequences", Proc. of FG'98, pp. 260-265, 1998.

[16]T. Mori, T. Kamisuwa, H. Mizoguchi, and T. Sato, "Action Recognition System based on Human Finder and Human Tracker", Proc. of IROS'97, pp.1334-1341, 1997.

[17]H. Inoue, T. Tachikawa and M. Inaba, "Robot Vision System with a Correlation Chip for Real-time Tracking, Optical Flow and Depth Map Generation", Proc. of ICRA'92, pp.1621-1626, 1992.

[18]H. Inoue, M. Inaba, T. Mori, and T. Tachikawa, "Real-Time Robot Vision System based on Correlation Technology", Proc. of ISIR, pp.675-680, 1993.

[19]T. Mori, M. Inaba and H. Inoue, "Visual Tracking based on Cooperation of Multiple Attention Regions", Proc. of ICRA'96, pp.2921-2928, 1996.

[20]T. Uchiyama, N. Sawasaki, T. Aoki, T. Morita, M. Sato, M., Inaba and H. Inoue, "Hardware Implementation of the Video-rate Tracking Vision", Proc. of the 12th Annual Conf. of the RSJ, pp.345-346, 1994.

[21]N. Sawasaki, T. Morita and T. Uchiyama, "Design and Implementation of High-speed Visual Tracking System for Real-time Motion Analysis", Proc. of the 13th ICPR, pp.478-483, 1996.

[22]T. Morita, N. Sawasaki, T. Uchiyama, and M. Sato, "Color Tracking Vision", Proc. of the 14th Annual Conference of the RSJ, pp.279-280, 1996.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Virtual Earphone to Whisper in a Person's Ear Remotely by Utilizing Visual Tracking and Speakers Array

Hiroshi Mizoguchi Takaomi Shigehara Yoshiyasu Goto Masashi Teshiba Taketoshi Mishima
Faculty of Engineering
Saitama University
255 Shimo Okubo, Urawa 338-8570, Japan

Abstract

This paper proposes a novel application of computer vision and signal processing, named Virtual Earphone. It integrates real-time face tracking and speakers array signal processing. By speakers array it is possible to form acoustic focus at the arbitrary location that is measured by the face tracking. Thus Virtual Earphone system can whisper in a person's ear as if an invisible virtual earphone were put by the person. The system is intended to be used as a voice output method for computer human interface, especially for autonomous intelligent agent that interacts with humans like as "digital secretary" or "digital butler". Utilizing the proposed Virtual Earphone, the agent is possible to clearly speak to human master remotely.

1 Introduction

Based upon rapid growth of computing power, expectation and demand for more "user friendly" computer that has advanced human interface and easily usable have been raised in recent years[1]-[8]. Especially a novel interface beyond traditional keyboard(KB) and display is demanded. Since traditional computer lacks "eyes", "ears" and "mouth", it cannot recognize phenomena in real world nor make influence to the world. People work and act in real world, not in front of KB and display, cannot enjoy usefulness and effectiveness of computers. If a computer were able to recognize our behavior visually and communicate with us via voice, it would be much convenient and useful. And moreover it could expand application fields and markets of computers.

In this paper the authors propose a novel computer-human interface, called Virtual Earphone. It integrates real-time face tracking and speakers array signal processing. By speakers array it is possible to form acoustic focus at the arbitrary location that is measured by the face tracking. Thus Virtual Earphone system can whisper in a person's ear as if an invisible virtual earphone were put by the person. The system is intended to be used as a voice output method for computer-human interface, especially for autonomous intelligent agent that interacts with humans like as "digital secretary" or "digital butler". Utilizing Virtual Earphone, the agent is possible to clearly speak to human master remotely.

The proposed system is an example of the above mentioned future "friendly" computer, even though the system lacks "ears", i.e. voice recognition capability. This study is our attempt to augment computer with eyes and mouth. To obtain benefits of the system, people do not need to sit in front of KB and display. The proposed system can be effectively used as a voice output method for HAL2000-like computing environment, such as Univ. Tokyo's Robotic Room[4], MIT AI Lab's Intelligent Room[2], MIT Media Lab's Smart Room[5], and so forth.

In the following, section 2 describes principle of the proposed Virtual Earphone. In section 3, real time face tracking is discussed. Section 4 describes mechanism of acoustic focusing by speakers array. Results of simulation are also described. Section 5 is conclusion.

2 Virtual Earphone

Fig. 1 illustrates principle of the proposed system. The system consists of face tracking part and speakers array part. The face tracking part detects and tracks human face. And it continuously calculates three dimensional coordinates of the face utilizing binocular stereo. To detect and track face area, skin color extraction method is utilized. Stereo matching is performed with dedicated hardware to calculate correlation. The calculated three dimensional coordinates is sent to speakers array part as output.

The speakers array part makes *acoustic focus* at the face location that is detected, tracked and measured by the

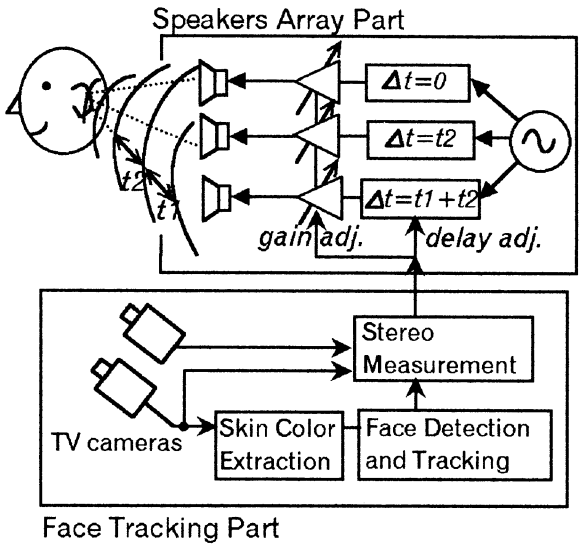


Fig. 1 Principle of Virtual Earphone

face tracking part. Since distances from a sound source to each speakers are different, a sound wave of the same phase reaches each speakers at different time. In other words, phases of the speakers output signal at a time are so different. The acoustic focus can be realized by equalizing the phases by adjusting delay and gain of each speakers output based upon the measured location of the face.

3 Face Tracking

The face tracking part consists of skin color extraction and binocular stereo. A method used in the skin color extraction part is based upon YIQ color representation instead of commonly used RGB one, because I and Q components of the YIQ are independent of brightness and free from brightness change. Original idea of the method is found in Mori et al[10]. Skin color is possible to be defined as some continuous region in IQ plane. Thus skin color area of the input image can be extracted by filtering each pixel of the input image based upon whether the corresponding point in the IQ plane is within the region or not.

The extracted face area in one eye is matched with corresponding area in other eye image for binocular stereo calculation. Stereo pair matching is based upon cross correlation utilizing dedicated hardware. The hardware is originally developed by Inoue et al. of Univ. of Tokyo [11][12][13] and commercialized by Fujitsu [14][15][16]. We utilize Fujitsu's product. It can calculate cross correlation of 8 by 8 template within search space at about 500 times during one video frame(1/30 sec). Since the calculation of cross correlation is performed at far beyond video rate by the hardware, the stereo calculation can be done in real time.

4 Speakers Array

This section describes principle of the acoustic focus by speakers array. Fig. 2 shows coordinate system of speakers array. As shown in the figure the speakers are laddered on x-axis. Each distance between speakers is denoted as d. In this discussion, frequency of sound is assumed as 1KHz, i.e. wavelength of the sound is about 34cm at ordinary room temperature. Distance between speakers, d, is set to half of wavelength, about 17cm. Speaker is indexed as Mi_i, where i ranges from $-N$ to N. Thus number of speakers is $2N+1$.

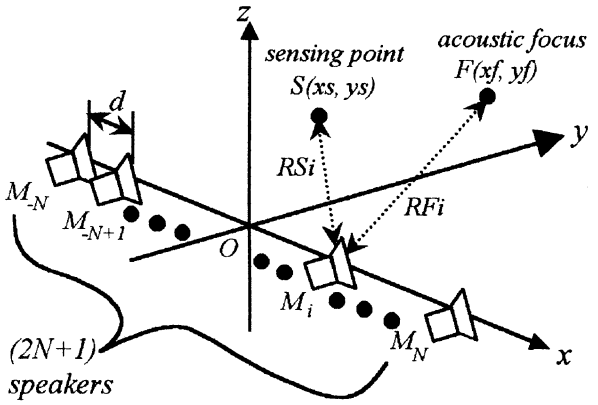


Fig. 2 Coordinate system of speakers array

In the figure, $F(xf, yf)$ denotes the acoustic focus. $S(xs, ys)$ denotes sensing point of the sound. Distance from the focus to i-th speaker is represented as RFi. Similarly RSi denotes distance from the sensing point to the i-th speaker. We assume that sound wave is spherically propagated from a speaker. If the sound from i-th speaker is sine wave with amplitude a_i and frequency f with delay b_i, it is written as

$$s_i(t) = a_i \sin 2\pi f(t - b_i) \qquad (1)$$

At the sensing point $S(xs, ys)$, composite sound is summation of sound elements propagated from each speakers. Here we write $xi(t)$ as the sound element from i-th speaker. The $xi(t)$ is written as follows.

$$xi(t) = \frac{a_i}{RSi} \sin 2\pi f (t - b_i - \tau_i) \qquad (2)$$

Where τi is the time delay from the i-th speaker to the sensing point. The delay is proportional to the distance from the speaker to the sensing point. The delay is written as

$$\tau i = RSi / v \qquad (3)$$

where v is velocity of sound at current temperature. The amplitude of the output from the i-th speaker decays in inverse relation to the distance from the speaker.

In case the sensing point is located at the focus point, there must be the furthest speaker from the focus and distance between the speaker and the focus is represented as *RFmax*. Putting additional delay to the output signal of each speaker by *(RFmax-RFi)/v* equalize phase of sound wave from each speaker's output at the sensing point *S(xs, ys)*. And setting gain of *i*-th speaker as *RFi* compensates the amplitude decaying. In other words, putting the additional delay by *(RFmax-RFi)/v* and setting gain as *RFi* to the *i*-th speaker realizes the acoustic focus at the location *F(xf, yf)*. The proposed Virtual Earphone sets the *F(xf, yf)* as the target person's face. Location of the face is obtained with real-time binocular stereo vision as described in the previous section.

The composite sound at the sensing point, *y(t)* is summation of sound elements from each speakers. It can be written as follows.

$$y(t) = \sum_{i=-N}^{N} xi(t) \qquad (4)$$

Thus this equation becomes

$$y(t) = (2N+1) \sin 2\pi f (t - \frac{RF\max}{v}) \qquad (5)$$

because of $RFi = RSi$.

In case the sensing point is not located at the focus point, i.e. $S(xs, ys) \neq F(xf, yf)$, the summation, *y(t)* is written as follows.

$$y(t) = \sum \frac{RFi}{RSi} \sin 2\pi f (t - \frac{RSi + RF\max - RFi}{v})$$

$$= \left(\sum_i \alpha i(xs, ys; xf, yf) \right) \sin 2\pi f t + \left(\sum_i \beta i(xs, ys; xf, yf) \right) \cos 2\pi f$$

$$= A(xs, ys; xf, yf) \sin(2\pi f t + B(xs, ys; xf, yf)) \qquad (6)$$

A(xs, ys; xf, yf) represents two dimensional spatial distribution of the amplitude for given focus *F(xf, yf)*.

To confirm the feasibility of the proposed idea, we have conducted computer simulation. Result of the simulation is shown in Fig. 3 and 4. These figures plot the spatial distribution of the amplitude for different focus points. In the figures, *z*-axis denotes decibel representation of the amplitude, *A(xs, ys ; xf, yf)*.

For better visibility, contour line is plotted on *x*-*y* plane. We can see higher amplitude spans from speakers (*x*-axis) to the focus point and the amplitude steeply decreases away from the focus.

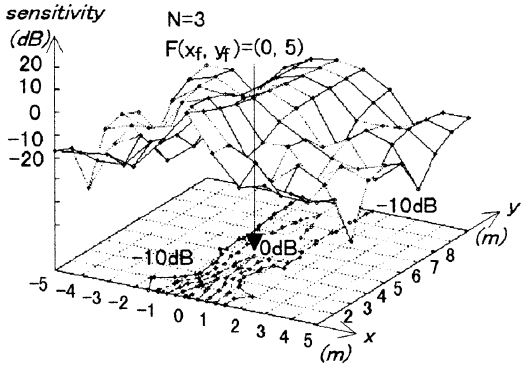


Fig. 3 Spatial distribution of amplitude (1)

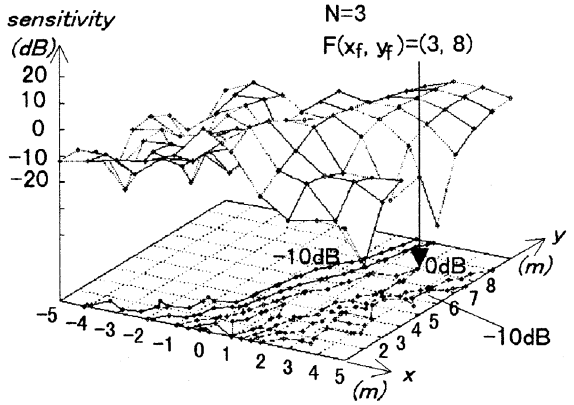


Fig. 4 Spatial distribution of amplitude (2)

Fig. 5 shows a result of different number of speakers. As the numbers increase, the amplitude distribution becomes steeper.

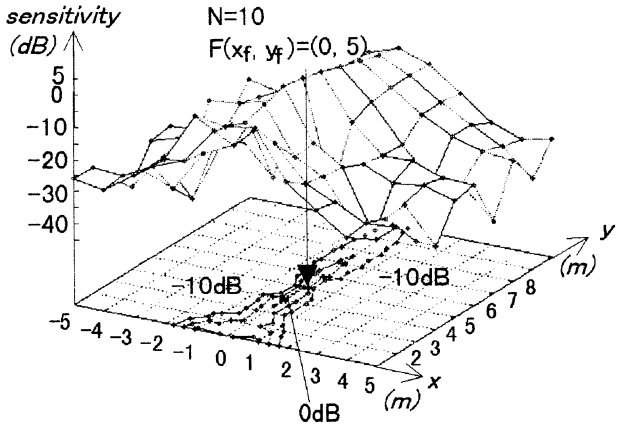


Fig. 5 Spatial distribution of amplitude (3)

5 Conclusion

This paper proposes a novel computer human interface, named Virtual Earphone. It integrates real time visual tracking of face and sound signal processing. Key technologies are real-time skin color extraction, correlation based stereo matching, and acoustic focusing with speakers array. Preliminary experiments and simulation results prove the feasibility of the proposed idea. Implementing a real system based upon the idea and experimenting by the system are future works.

References

[1] P. Maes, "Agents that reduce work and information overload", *Communications of the ACM*, Vol.37, No.7, pp.31-40, 1994.

[2] M. C. Torrance, "Advances in Human-Computer Interaction: The Intelligent Room", *Working Notes of the CHI95 Research Symposium*, 1995.

[3] T. Sato, Y. Nishida and H. Mizoguchi, "Robotic room: Symbiosis with human through behavior media", *Robotics and Autonomous Systems*, No.18, pp.185-194, 1996.

[4] H. Mizoguchi, T. Sato and T. Ishikawa, "Robotic office room to support office work by human behavior understanding function with networked machines", *IEEE/ASME Transaction on Mechatronics*, Vol.1, No.3, pp.237-244, 1996.

[5] A. Pentland, "Smart Rooms", *Scientific American*, pp.54-62, 1996.

[6] H. Asada and I.W. Hunter, "Total Home Automation and Health Care/Elder Care", *Technical Report, Department of Mechanical Engineering*, MIT, 1996.

[7] R. P. Picard, *Affective Computing*, MIT Press, 1997.

[8] S. Mann, "Wearable computing: A first step toward personal imaging", *Computer*, pp.25-31, 1997.

[9] W. Dai and K. Sasaki, "Basic Study for Realizability of a Telescopic Microphone System", *Transactions of SICE*, Vol.35, No.8, pp.843-845, 1997.

[10]T. Mori, T. Kamisuwa, H. Mizoguchi, and T. Sato, "Action Recognition System based on Human Finder and Human Tracker", *Proceedings of IROS'97*, pp.1334-1341, 1997.

[11]H. Inoue, T. Tachikawa and M. Inaba, "Robot Vision System with a Correlation Chip for Real-time Tracking, Optical Flow and Depth Map Generation", *Proceedings of ICRA'92*, pp.1621-1626, 1992.

[12]H. Inoue, M. Inaba, T. Mori, and T. Tachikawa, "Real-Time Robot Vision System based on Correlation Technology", *Proceedings of ISIR*, pp.675-680, 1993.

[13]T. Mori, M. Inaba and H. Inoue, "Visual Tracking based on Cooperation of Multiple Attention Regions", *Proceedings of ICRA'96*, pp.2921-2928, 1996.

[14]T. Uchiyama, N. Sawasaki, T. Aoki, T. Morita, M. Sato, M., Inaba and H. Inoue, "Hardware Implementation of the Video-rate Tracking Vision", *Proceedings of the 12th Annual Conference of the Robotics Society of Japan*, pp.345-346, 1994.

[15]N. Sawasaki, T. Morita and T. Uchiyama, "Design and Implementation of High-speed Visual Tracking System for Real-time Motion Analysis", *Proceedings of the 13th International Conference on Pattern Recognition*, pp.478-483, 1996.

[16]T. Morita, N. Sawasaki, T. Uchiyama, and M. Sato, "Color Tracking Vision", *Proceedings of the 14th Annual Conference of the Robotics Society of Japan*, pp.279-280, 1996.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Virtual Concierge: A Talking Door-phone to Speak up Visitor's Name

Hiroshi Mizoguchi Takaomi Shigehara Yoshiyasu Goto Taketoshi Mishima
Faculty of Engineering
Saitama University
255 Shimo Okubo, Urawa 338-8570, Japan

Abstract

This paper proposes a talking door-phone system as a novel application of human computer interface(HCI) that integrates face image processing and speech sound processing. The system consists of a CCD camera attached on a door, a loud speaker installed in a room, and a computer connected with both the camera and the speaker. The system recognizes and identifies a visitor's face by image processing. If the face is known and identification succeeds, the system speaks up the visitor's name inside the door by using text-to-speech synthesizer. In case that the face is not previously registered or recognition fails, the system reports the impossibility of identification with the synthesized voice. Thus the system works as if an invisible virtual concierge were on stand-by at the door.

1 Introduction

Rapid growth of computing power in recent years has raised expectation and demand for more "user friendly" computer that has advanced human interface and is easily manipulated. Especially a novel interface beyond traditional keyboard(KB) and display is demanded. Since traditional computer lacks "eyes", "ears" and "mouth", it is blind and deaf. That is the computer cannot recognize phenomena happened in real world nor make influence to the real world. Thus, people work and act in the real world, not in front of KB and display, cannot enjoy usefulness and effectiveness of computers. If a computer were able to recognize our behavior visually and communicate with us via voice, it would be much more effective, convenient and useful. And it would expand application fields of computers.

As an effort to add eye and mouth to computers, this paper proposes a talking door-phone system, named Virtual Concierge. The system consists of a CCD camera attached on a door, a loud speaker installed in a room, and a computer connected with both the camera and the speaker. The system recognizes and identifies a visitor's face by image processing. If the face is known and identification succeeds, the system speaks up the visitor's name inside the door by using text-to-speech synthesizer. The background aim of this study is to explore required functions to realize such future "friendly" computer and methods to meet the requirements. The proposed system is an example image of such future computer, even though the system lacks "ears", i.e. voice recognition capability.

Based upon recent progress in face recognition technology[9]-[15], increase of computing power, increased quality of voice synthesizer, and demand for more sophisticated human interface, it is possible and significant to implement a functional prototype of the proposed talking door-phone system. This implementation will prove feasibility of the proposed idea and contribute to examine required functions. The authors are currently implementing the functional prototype.

The functional prototype system utilizes face detection, face recognition, and software speech synthesizer. Each modules are implemented as independent processes. They are connected each other by socket interface using internet protocol (IP). Thus each parts and whole system of the prototype are all internet reachable. It contributes that these parts are easily applicable to other applications. It also contributes easy expansion and maintenance of the system.

In the following, section 2 describes principle of the proposed system. Section 3 and section 4 describe the currently implementing functional prototype system. Section 5 is conclusion

2 Virtual Concierge

Fig. 1 shows a block diagram of the proposed talking door-phone system. This figure illustrates both principle of the proposed system and typical situation where the system is used. The system consists of face detection part, recognition part, and text-to-speech synthesizer. As the input and output devices, a CCD camera and a loud speaker are utilized. The face detection and recognition

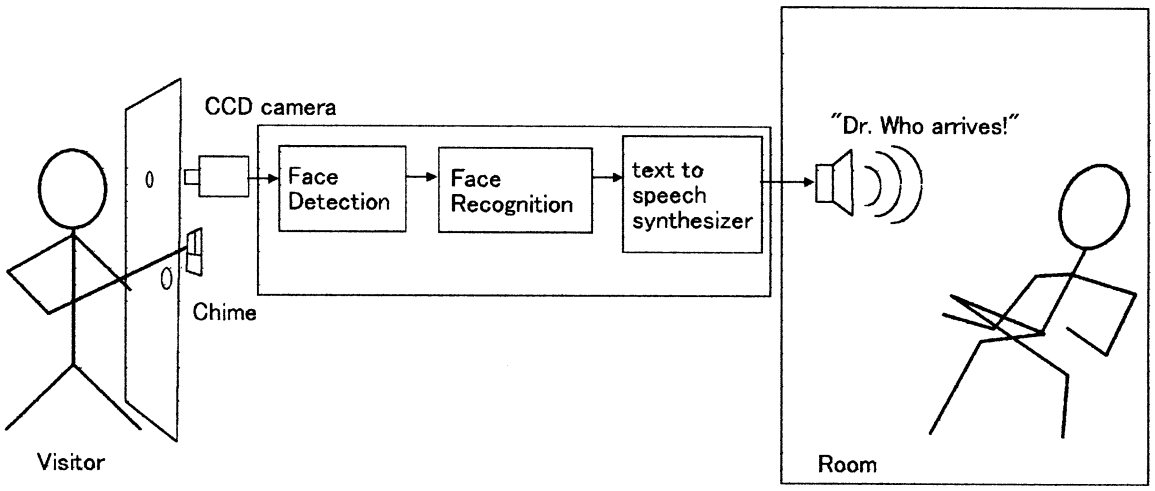


Fig. 1 Block Diagram of Virtual Concierge System

part process input face image continuously. To detect and track face area, skin color extraction method is utilized. To identify input face image, the face recognition part utilize cross correlation. If the recognition succeeds, its result is sent to the text-to-speech synthesizer part. This part converts input text to synthesized voice and drives the loud speaker. As the result, the system speaks up the visitor's name inside the door.

Fig. 2 shows structure of currently implementing prototype system. On a PC running Linux, the face detecion and recognition parts are implemented as processes. The text-to-speech synthesizer runs as a process on the other PC running Windows95. These two PCs are connected through LAN(ethernet). Thus the face detection, recognition and speech synthesizer are connected via IP protocol using "socket" interface.

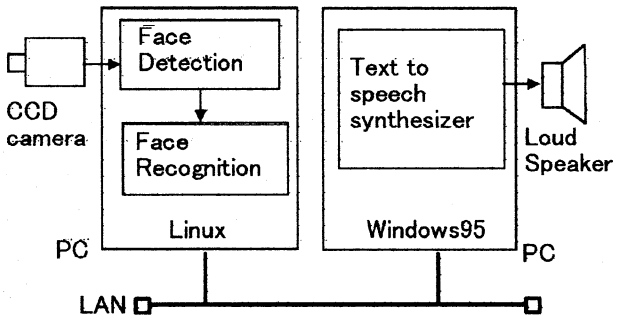


Fig. 2 Structure of Current Prototype

3 Face Detection and Recognition

The face detection is realized by skin color extraction. A method to extract skin color in the face detection part is based upon YIQ color representation instead of commonly used RGB one. The same method is utilized in [16]. Since I and Q components of the YIQ are independent of brightness, the method is robust against brightness change. Original idea of the method is found in Mori et al.[17].

Skin color is possible to be defined as some continuous region in the IQ plane of the YIQ representation. Thus skin color area of the input image can be extracted by filtering each pixel of the input image based upon whether the corresponding point in the IQ plane is within the region or not. According to preliminary experiment, quite similar extracted areas can be obtained while brightness level of input images are so different.

The face recognition is performed to the extracted skin color area. The extracted area is a candidate of a face image. To recognize the input face image, a pattern matching process is performed within the extracted area to find whether the candidate is matched to previously registered face images or not. The matching process is based upon cross correlation. To calculate the cross correlation a dedicated hardware is utilized.

The dedicated hardware to calculate to cross correlation of images is originally developed by Inoue et

al. of Univ. of Tokyo[18][19][20] and commercialized by Fujitsu [21][22] [23]. The authors utilize Fujitsu's product. It enables to calculate cross correlation of 8 by 8 template within search space at about 500 times during one video frame (1/30 sec). Since the calculation of cross correlation is performed at far beyond video rate by the hardware, the recognition process can be done in real time.

4 Speech Synthesis

The text-to-speech synthesizer used in the prototype system is named Smart Talk made by OKI Electric. Smart Talk is not a hardware synthesizer but a software package for Window95. It includes not only application programs but also library (DLL). The DLL is callable from Visual Basic(VB), Visual C(VC) and other Microsoft's software products. Thus we can develop our own application programs using the DLL. That is the reason why we have chosen Smart Talk as the speech synthesizer of the prototype system.

Fig. 3 shows software structure of the speech synthesizer of the prototype system. In the figure, winsock library is a part of Window95 operating system

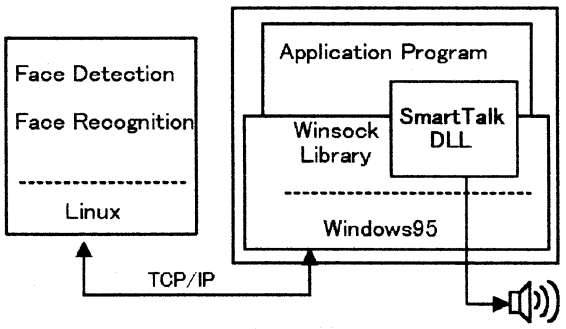


Fig. 3 Software Structure of Speech Synthesizer

and provides "socket" interface to application programs. It can be called from VB, VC and so forth. The application program is written in VB. It receives results from the face recognition part via winsock library and generates texts to speak up based upon the recognition results.

The implementing prototype has no capability to learn and correct when the system fails the recognition. In case of human concierge, human master is possible to correct wrong recognition by simply telling it. On the contrary, the current prototype has no function to recognize human voice and cannot understand what the master says. Thus, instead of correction by voice, the human master corrects wrong recognition and/or teach unknown face to the system interactively using interactive nature of VB.

5 Conclusion

This paper proposes a novel application of face image processing to a computer human interface. The proposed system utilizes a CCD camera attached on a door, a loud speaker in a room, and computers connected with the camera and the speaker. Software peech synthesizer is running on the computer. The system recognizes input face image and speaks up the name inside the door by using the speech synthesizer. The proposed system is useful and because it works as if it were an invisible virtual concierge having good memory for faces and names of visitors.

At the time of writing this paper, the authors are implementing a functional prototype of the proposed system, using currently commercially available components. In this paper, implementation issues of the prototype are also described. Integrating elemental technologies, completing the prototype, conducting experiments and evaluating the prototype are future works.

References

[1] P. Maes, "Agents that reduce work and information overload", CACM, Vol.37, No.7, pp.31-40, 1994.

[2] M. C. Torrance, "Advances in Human-Computer Interaction: The Intelligent Room", Working Notes of the CHI95 Research Symposium, 1995.

[3] A. Pentland, "Smart Rooms", Scientific American, pp.54-62, 1996.

[4] B. J. Rhodes and T. Starner, "Remembrance agent: A continuously running automated information retrieval system", Proc. of First Int'l Conf. on the Practical Application of Intelligent Agents and Multi Agent Technology, pp. 487-495, 1996.

[5] R. W. Picard, Affective Computing, MIT Press, 1997.

[6] S. Mann, "Wearable computing: A first step toward personal imaging", Computer, pp.25-31, 1997.

[7] T. Starner, J. Weaver, and A. Pentland, "A wearable computer based American sign language recognizer", Proc. of the First ISWC, 1997.

[8] R. W. Picard and J. Healey, "Affective wearables", Proc. of the First ISWC, 1997.

[9] M. Turk and A. Pentland, "Face Recognition Using Eigenfaces", Proc. of CVPR'91, pp.586-591, 1991.

[10]T. Kurita, N. Otsu, and T. Sato, "A Face Recognition Method using Higher Order Local Autocorrelation and Multivariate Analysis", Proc. of ICPR'92, pp. 213-216, 1992.

[11]I. Craw, N. Costen, T. Kato, G. Robertson, and S. Akamatsu, "Automatic Face Recognition: Combining Configuration and Texture", Proc. of FG'95, pp. 53-58, 1995.

[12]T. Sakaguchi and S. Morishima, "Face feature extraction from spatial frequency for dynamic expression recognition", Proc. of ICPR'96, pp. 451-455, 1996.

[13]T. Kurita, K. Hotta and T. Mishima, "Scale and Rotation Invariant Recognition Method using Higher-Order Local Autocorrelation Features of Log-Polar Image, Proc. of ACCV'98, 1998.

[14]S. A. Rizvi, P. J. Phillips and H. Moon, "The FERET Verification Testing Protocol for Face Recognition Algorithms", Proc. of FG'98, pp.260-265, 1998.

[15]G. J. Edwards, C. J. Taylor and T. F. Cootes, "Learning to Identify and Track Faces in Image Sequences", Proc. of FG'98, pp. 260-265, 1998.

[16]H. Mizoguchi, T. Shigehara, Y. Goto, and T. Mishima, "A Novel Application of Face Image Processing to Wearable Computer - Augmentation of Human Memory for Faces and Names -", Proc. of AROB'99, 1999.

[17]T. Mori, T. Kamisuwa, H. Mizoguchi, and T. Sato, "Action Recognition System based on Human Finder and Human Tracker", Proc. of IROS'97, pp.1334-1341, 1997.

[18]H. Inoue, T. Tachikawa and M. Inaba, "Robot Vision System with a Correlation Chip for Real-time Tracking, Optical Flow and Depth Map Generation", Proc. of ICRA'92, pp.1621-1626, 1992.

[19]H. Inoue, M. Inaba, T. Mori, and T. Tachikawa, "Real-Time Robot Vision System based on Correlation Technology", Proc. of ISIR, pp.675-680, 1993.

[20]T. Mori, M. Inaba and H. Inoue, "Visual Tracking based on Cooperation of Multiple Attention Regions", Proc. of ICRA'96, pp.2921-2928, 1996.

[21]T. Uchiyama, N. Sawasaki, T. Aoki, T. Morita, M. Sato, M., Inaba and H. Inoue, "Hardware Implementation of the Video-rate Tracking Vision", Proc. of the 12th Annual Conf. of the RSJ, pp.345-346, 1994.

[22]N. Sawasaki, T. Morita and T. Uchiyama, "Design and Implementation of High-speed Visual Tracking System for Real-time Motion Analysis", Proc. of the 13th ICPR, pp.478-483, 1996.

[23]T. Morita, N. Sawasaki, T. Uchiyama, and M. Sato, "Color Tracking Vision", Proc. of the 14th Annual Conference of the RSJ, pp.279-280, 1996.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Realizing Virtual Wireless Microphone to Pick up Human Voice Remotely and Clearly

Hiroshi Mizoguchi Takaomi Shigehara Masashi Teshiba Taketoshi Mishima

Faculty of Engineering
Saitama University
255 Shimo Okubo, Urawa 338-8570, Japan

Abstract

This paper proposes a novel application of computer vision and signal processing, named Virtual Wireless Microphone(VWM). It integrates real-time face tracking and sound signal processing. VWM is intended to be used as a speech signal input method for human computer interaction(HCI), especially for autonomous intelligent agent that interacts with humans like as "digital secretary". Utilizing VWM, the agent can clearly listen human master's voice remotely as if a wireless microphone was put just in front of the master.

1 Introduction

Both semiconductor technology and computing power have rapidly grown in these years. These remarkable trends have raised an expectation for more friendly computer-human interface[1]-[8]. If computer could listen human voice command and behave properly, it would be very convenient and much desirable for us. For example, an autonomous intelligent agent interacts with humans like as a *digital secretary*. But even state-of-the-art speech recognition technology requires high quality noiseless input. User of current speech recognition system must put headset microphone to prevent background noise. Because of this difficulty of usage the speech recognition still cannot be applied to human computer interaction.

To make the speech recognition practically applicable to the human computer interaction, some novel technique to pick up human voice clearly and remotely is keenly required. In other words, a technique to form *acoustic focus* at human face is needed. To realize such technique there are two problems to be solved. One is how to track and measure location of the face in real time. The other is how to form the acoustic focus at the measured location. The location of the human face is not fixed according to motion of the human's head and body. Especially, in case of the digital secretary application, the user may walk around in his/her office room.

To solve these problems this paper proposes a novel technique to form acoustic focus at the human face by integrating computer vision and signal processing. The proposed technique is named Virtual Wireless Microphone(VWM). As for the first problem, real time face tracking and binocular stereo vision are utilized. Face tracking is realized with skin color extraction. Binocular stereo vision calculates three dimensional location of the tracked face. As for the second problem, the proposed technique utilizes microphones array. Setting gain and delay of each microphone properly enables to form acoustic focus at desired location. Based upon the face location measured by the above mentioned stereo vision, gain and delay of each microphone are determined. By repeating this calculation at video rate enables the acoustic focus to track the human face in real time. Dai et al.[9] discussed microphones array as a mean to realize telescopic microphone system with variable focus. Our work reported in this paper is an extension of their work by integrating with real-time computer vision.

In the following, section 2 describes principle of the proposed virtual wireless microphone. In section 3, real time face tracking is discussed. Section 4 describes mechanism of acoustic focusing by microphones array. Results of simulation are also described. Section 5 discusses implementation issues. Section 6 is conclusion.

2 Virtual Wireless Microphone

Fig. 1 shows principle of the proposed system. The system consists of face tracking part and microphones array part. The face tracking part detects and tracks the face. And it continuously calculates three dimensional coordinates of the face utilizing binocular stereo. To detect and track face area, skin color extraction method is utilized. Stereo matching is performed with dedicated hardware to calculate correlation. The calculated three

dimensional coordinates is sent to microphones array part as output.

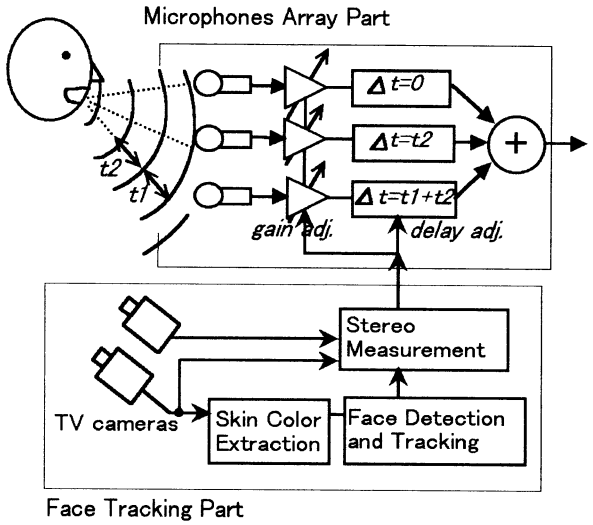


Fig. 1 Principle of proposed system

The microphones array part makes *acoustic focus* at the face location that is detected, tracked and measured by the face tracking part. Since distances from a sound source to each microphones are different, a sound wave of the same phase reaches each microphones at different time. In other words, phases of the microphones output signal at a time are so different. The acoustic focus can be realized by equalizing the phases by adjusting delay and gain of each microphones output based upon the measured location of the face.

3 Face Tracking

The face tracking part consists of skin color extraction and binocular stereo. A method used in the skin color extraction part is based upon YIQ color representation instead of commonly used RGB one, because I and Q components of the YIQ are independent of brightness and free from brightness change. Original idea of the method is found in Mori et al[10]. Skin color is possible to be defined as some continuous region in IQ plane. Thus skin color area of the input image can be extracted by filtering each pixel of the input image based upon whether the corresponding point in the IQ plane is within the region or not.

The extracted face area in one eye is matched with corresponding area in other eye image for binocular stereo calculation. Stereo pair matching is based upon cross correlation utilizing dedicated hardware. The hardware is originally developed by Inoue et al. of Univ. of Tokyo [11][12][13] and commercialized by Fujitsu [14][15][16]. We utilize Fujitsu's product. It can calculate cross correlation of 8 by 8 template within search space at about 500 times during one video frame(1/30sec). Since the calculation of cross correlation is performed at far beyond video rate by the hardware, the stereo calculation can be done in real time. Fig. 2 shows an example of stereo pair matching done with the hardware.

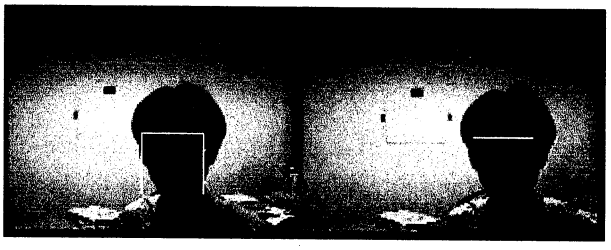


Fig. 2 Stereo pair matching example

4 Microphones Array

This section describes principle of the acoustic focus. Fig. 3 shows coordinate system of microphones array. As shown in the figure the microphones are laddered on *x*-axis. Each distance between microphones is denoted as *d*. In this discussion, frequency of sound is assumed as 1KHz, i.e. wavelength of the sound is about 34cm at ordinary room temperature. Distance between microphones, *d*, is set to half of wavelength, about 17cm. Microphone is indexed as Mi_i, where *i* ranges from -*N* to *N*. Thus number of microphones is 2*N*+1.

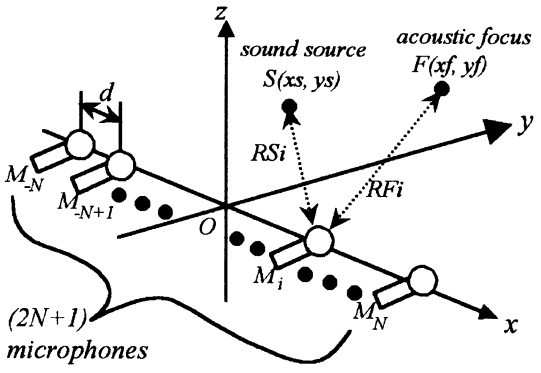


Fig. 3 Coordinate system of microphones array

In the figure, $F(xf, yf)$ denotes the acoustic focus. $S(xs, ys)$ denotes the sound source. Distance from the focus to i-th microphone is represented as RFi. Similarly RSi denotes distance from the sound source to i-th microphone. When the sound source is sine wave with amplitude 1 and frequency f, it is written as

$$s(t)=\sin 2\pi ft \qquad (1)$$

Output of i-th microphone is as follows.

$$xi(t)=\frac{1}{RSi}\sin 2\pi f(t-\tau i) \qquad (2)$$

Where τi is the time delay from the source to the i-th microphone. The delay is proportional to the distance from the source to the microphone. The delay is written as

$$\tau i = RSi / v \qquad (3)$$

where v is velocity of sound at current temperature.
The amplitude of the output from the i-th microphone decays in inverse relation to the distance from the sound source.

In case the sound source is located at the focus point, there must be the furthest microphone from the focus and distance between the microphone and the focus is represented as $RFmax$. Putting additional delay to the output signal of each microphone by $(RFmax-RFi)/v$ equalize phase of each microphone's output. And setting gain of i-th microphone as RFi compensates the amplitude decaying. In other words, putting the additional delay by $(RFmax-RFi)/v$ and setting gain as RFi realizes the acoustic focus at the location $F(xf, yf)$. Location of the targeted face is obtained with real-time binocular stereo vision as described in the previous section. Summation of each microphone's output $y(t)$ can be written as;

$$y(t)=\sum_{i=-N}^{N} RFixi(t) \qquad (4)$$

$$y(t)=(2N+1)\sin 2\pi f(t-\frac{RF\max}{v}) \qquad (5)$$

because of $RFi = RSi$.

In case the sound source is not located at the focus point, i.e. $S(xs, ys) \neq F(xf, yf)$, the summation, $y(t)$ is written as follows.

$$y(t)=\sum\frac{RFi}{RSi}\sin 2\pi f(t-\frac{RSi+RF\max-RFi}{v})$$

$$=\left(\sum_i \alpha i(xs, ys; xf, yf)\right)\sin 2\pi ft$$

$$+\left(\sum_i \beta i(xs, ys; xf, yf)\right)\cos 2\pi ft$$

$$=A(xs, ys; xf, yf)\sin(2\pi ft+B(xs, ys; xf, yf)) \qquad (6)$$

$A(xs, ys; xf, yf)$ represents the sensitivity of the microphones array for given focus $F(xf, yf)$.

To confirm the feasibility of the proposed idea, we have conducted computer simulation. Result of the simulation is as follows. Fig. 5 plots the spatial distribution of the sensitivity for different focus points. In the figures, z-axis denotes decibel representation of the sensitivity, $A(xs, ys ; xf, yf)$. For better visibility, contour line is plotted on x-y plane. We can see higher sensitivity spans from microphones (x-axis) to the focus point and the sensitivity steeply decreases away from the focus.

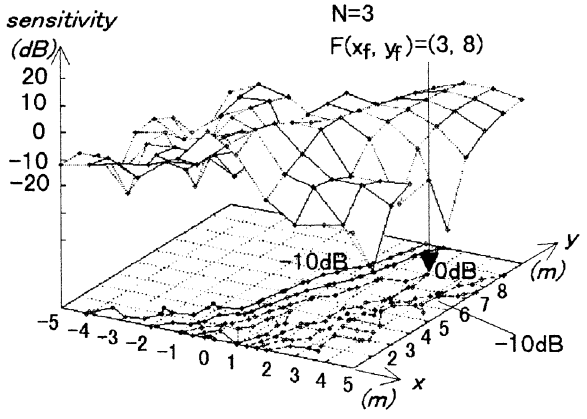


Fig. 5 Spatial distribution of sensitivity

5 Implementation

Since the results of software simulation prove feasibility of the proposed idea, the authors are implementing a real hardware. As described above, the Fujitu's tracking vision is utilized for both tracking and stereo matching in the face tracking part in the system. A color video capture card made by Argo Craft, Tokyo, Japan, is also utilized in the part to extract skin color. The reason to choose the card is that there are driver software for not only Windows NT/95 but also Linux and FreeBSD. Moreover Argo Craft distributes source codes of these driver and sample programs for free.

As for the microphones array part, the authors are now evaluating microprocessor for the sound signal processing. Sampling frequency is planned as 44.1KHz in order to realize CD sound quality. In this case, the sampling period is 22.7 micro seconds. A/D conversion,

multiply for setting gain, access to tapped queue for setting delay, summation of all microphones outputs should be completed within the period, 22.7 micro seconds.

Conventionally DSPs are applied to such kind of task. However we are considering Hitachi's SH-x family of RISC based embedded microprocessor instead of DSPs. There are several reasons. One is the processor includes fast 8 channel, 10bit A/D converter. Conversion time is 6.7 micro seconds. And 42 bit accumulation of 16 bit by 16 bit multiply can be executed in only 0.1-0.15 micro second. The other reason is the SH-x has simple and normal architecture different from peculiar one often found in DSPs. Consequently it is easy to program and debug. It has rich programming support tools, such as cross C compiler, assembler, linker, debugger, profiler, and so forth. Moreover its power consumption is quite small.

6 Conclusion

This paper proposes a novel technique to form acoustic focus at human face dynamically. It integrates real time face tracking and microphones array signal processing. Key technologies are real-time skin color extraction, correlation based stereo matching, and acoustic focusing with microphones array. Preliminary experiments and simulation results support the feasibility of the proposed idea. Implementing a real system based upon the idea and experimenting by the system are future works.

References

[1] P. Maes, "Agents that reduce work and information overload", CACM, Vol.37, No.7, pp.31-40, 1994.

[2] M. C. Torrance, "Advances in Human-Computer Interaction: The Intelligent Room", Working Notes of the CHI95 Research Symposium, 1995.

[3] T. Sato, Y. Nishida and H. Mizoguchi, "Robotic room: Symbiosis with human through behavior media", Robotics and Autonomous Systems, No.18, pp.185-194, 1996.

[4] H. Mizoguchi, T. Sato and T. Ishikawa, "Robotic office room to support office work by human behavior understanding function with networked machines", IEEE/ASME Transaction on Mechatronics, Vol.1,No.3, pp.237-244, 1996.

[5] A. Pentland, "Smart Rooms", Scientific American, pp.54-62, 1996.

[6] H. Asada and I.W. Hunter, "Total Home Automation and Health Care/Elder Care", Technical Report, Department of Mechanical Engineering, MIT, 1996.

[7] R. P. Picard, Affective Computing, MIT Press, 1997.

[8] S. Mann, "Wearable computing: A first step toward personal imaging", Computer, pp.25-31, 1997.

[9] W. Dai and K. Sasaki, "Basic Study for Realizability of a Telescopic Microphone System", Transactions of SICE, Vol.35, No.8, pp.843-845, 1997.

[10] T. Mori, T. Kamisuwa, H. Mizoguchi, and T. Sato, "Action Recognition System based on Human Finder and Human Tracker", Proceedings of IROS'97, pp.1334-1341, 1997.

[11] H. Inoue, T. Tachikawa and M. Inaba, "Robot Vision System with a Correlation Chip for Real-time Tracking, Optical Flow and Depth Map Generation", Proceedings of ICRA'92, pp.1621-1626, 1992.

[12] H. Inoue, M. Inaba, T. Mori, and T. Tachikawa, "Real-Time Robot Vision System based on Correlation Technology", Proceedings of ISIR, pp.675-680, 1993.

[13] T. Mori, M. Inaba and H. Inoue, "Visual Tracking based on Cooperation of Multiple Attention Regions", Proceedings of ICRA'96, pp.2921-2928, 1996.

[14] T. Uchiyama, N. Sawasaki, T. Aoki, T. Morita, M. Sato, M., Inaba and H. Inoue, "Hardware Implementation of the Video-rate Tracking Vision", Proceedomgs of the 12th Annual Conference of the Robotics Society of Japan, pp.345-346, 1994.

[15] N. Sawasaki, T. Morita and T. Uchiyama, "Design and Implementation of High-speed Visual Tracking System for Real-time Motion Analysis", Proceedings of the 13th ICPR, pp.478-483, 1996.

[16] T. Morita, N. Sawasaki, T. Uchiyama, and M. Sato, "Color Tracking Vision", Proceedings of the 14th Annual Conference of the Robotics Society of Japan, pp.279-280, 1996.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Neural Network Based Artificial Life Model for Navigation of Multiple Autonomous Mobile Robots in the Dynamic Environment

Suk-Ki Min and Hoon Kang
Dept. of Control and Instrumentation Eng. Chung-Ang Univ.
221, Heuksuk-Dong, Dongjak-Gu, Seoul, 156-756 Korea
Tel:+82-2-820-5320 Fax:+82-2-816-1856 E-mail:hkang@cau.ac.kr

Abstract: The objective of this paper is, based upon the principles of artificial life, to induce emergent behaviors of multiple autonomous mobile robots which complex global intelligence form from simple local interactions. Here, we propose architecture of neural network learning with reinforcement signals which perceives the neighborhood information and decides the direction and the velocity of movement as mobile robots navigates in a group. As the results of the simulations, the optimum weight is obtained in real time, which not only prevent the collisions between agents and obstacles in the dynamic environment, but also have the mobile robots move and keep in various patterns.

Keywords: artificial life, autonomous mobile robots, neural network, flocking, obstacle avoidance

I. Introduction

Autonomous mobile robot systems(AMRS) achieve the desired task by perceiving the neighborhood status and by showing appropriate behaviors from the collected sensor data. Each individual system has an independent control structure which reacts in an orderly fashion and therefore, it can be endowed with an efficient function of collectively cooperative and common tasks. Brooks[1] suggested a simple control structure that performs a given task.

Recent works on multiple autonomous mobile robots are focused upon individual and local interactions between robot agents. Artificial life[2] deals with local interactions between agents with simpler structures resulting in global behaviors, and it places emphasis upon the fact that collective intelligence is rather induced from such interactions than obtained from the results based on artificial intelligence techniques. Related works on collective intelligence and emergent behaviors are also given in [3-5]

In this paper, we will show that complex behaviors result from the simple local interactions among autonomous mobile robots. The control structure of autonomous mobile robots consists of reconfigurable neural networks in which the weights are updated independently according to the dynamic environment. Therefore, these weights are different from each other due to various conditions in the neighborhoods as they continue to change.

II. Action Network

Mobile robots collect sensor-input data that represent relative position vectors of other robots and/or obstacles. each sensor has the sensing area in which the robot detects other agents within the front 180° range. The sensing area is also composed of two subregions, the flocking area and the threat

area. The output variables of the mobile robots are the robot velocity that can be decomposed into the magnitude $|\mathbf{v}_i|$ and the normalized direction vector $\mathbf{d}_i/|\mathbf{d}_i|$ and the angle with which the robot turns.

1. Strategy for Robot Velocity

We choose acceleration if the distance between one and others is large or deceleration if that is small. If $\Delta s(t)$ is defined as the distance to add or subtract as the robot proceeds while maintaining the desired distance, the new speed can be obtained from the acceleration- deceleration scheme using $\Delta s(t)$.

2. Strategies of Selecting Direction Vectors

Based on the direction vector $\mathbf{d}_i$, we define the right vector $[\mathbf{A}]_i^+$ if an arbitrary relative position vector $\mathbf{A}_i$ is on the righthand side of the i-th robot's sensing area, and the left vector $[\mathbf{A}]_i^-$ if it is on the lefthand side.

2.1 Strategy for Flocking Behaviors

$$\mathbf{d}_i(t+1) = \sum_{j=0}^{N-1} (w_i^a(t) \cdot \mathbf{P}_{ij}^a(t) + \mathbf{O}_{ij}^a(t))$$

$$\mathbf{O}_{ij}^a(t) = \begin{cases} [\mathbf{O}_{ij}^a(t)]_i^+ & \text{if } \mathbf{P}_{ij}^a(t) = [\mathbf{P}_{ij}^a(t)]_i^- \\ [\mathbf{O}_{ij}^a(t)]_i^- & \text{if } \mathbf{P}_{ij}^a(t) = [\mathbf{P}_{ij}^a(t)]_i^+ \end{cases}$$

$$\mathbf{P}_{ij}^a(t) \cdot \mathbf{O}_{ij}^a(t) = 0 \quad (|\mathbf{O}_{ij}^a(t)| = 1) \qquad (1)$$

where $w_i^a(t)$ is the weight of the robot, $\mathbf{P}_{ij}^a(t)$ is the position vector of the i-th robot indicating the j-th other robot within the sensing area at time t, $\mathbf{O}_{ij}^a(t)$ is an orthogonal unit vector with respect to $\mathbf{P}_{ij}^a(t)$, and N is the number of robots in the sensing area. The larger the weight values are, the easier each robot is ready to follow the neighborhood robots. If not, each robot may avoid other robots. Thus, the selection of appropriate values for $w_i^a(t)$ is important.

2.2 Obstacle Avoidance Strategy

$$\mathbf{d}_i(t+1) = \min_{j=0}^{M-1} (w_i^o(t) \cdot \mathbf{P}_{ij}^o(t) + \mathbf{O}_{ij}^o(t))$$

$$\mathbf{O}_{ij}^o(t) = \begin{cases} [\mathbf{O}_{ij}^o(t)]_i^+ & \text{if } OB(t) = 1 \\ [\mathbf{O}_{ij}^o(t)]_i^- & \text{if } OB(t) = -1 \\ [\mathbf{O}_{ij}^o(t)]_i^+ & \text{if } OB(t) = 0 \ \& \\ & \mathbf{P}_{ij}^o(t) = [\mathbf{P}_{ij}^o(t)]_i^- \\ [\mathbf{O}_{ij}^o(t)]_i^- & \text{if } OB(t) = 0 \ \& \\ & \mathbf{P}_{ij}^o(t) = [\mathbf{P}_{ij}^o(t)]_i^+ \end{cases}$$

$$\mathbf{P}_{ij}^o(t) \cdot \mathbf{O}_{ij}^o(t) = 0 \quad (|\mathbf{O}_{ij}^o(t)| = 1) \qquad (2)$$

where $\min_{i=0}^{n} \mathbf{a}_i$ is the operator that returns the minimum magnitude vector among $\mathbf{a}_i$, $w_i^o(t)$ is the weight of an obstacle (boundary), $\mathbf{P}_{ij}^o(t)$ is the position vector indicating the j-th obstacle within the sensing area of the i-th robot, $\mathbf{O}_{ij}^o(t)$ is an orthogonal unit vector with respect to $\mathbf{P}_{ij}^o(t)$, and M is the number of obstacles within the sensing area.

If OB=1 or OB=-1, the orthogonal unit vector becomes either right or left vector regardless of the position vector of the obstacle (boundary). If OB=0, the orthogonal unit vector becomes either right or left vector according as the position vector of the obstacle (boundary) is either right or left vector. And, OB becomes 1 or -1 according as the direction vector in the next step is either right or left vector. OB becomes 0 if the robot passes the obstacle (boundary) completely or encounters another obstacle (boundary). The orientation bits have effects only on the obstacle avoidance mode and will move the robot safely and efficiently as it faces obstacles or boundaries.

2.3 Collision Avoidance Strategy

We define the danger ratio in order to select the most safe direction as follows:

$$\text{danger ratio} = 1 - |\mathbf{P}_{ij}|/TR \qquad (3)$$

where TR is the threat radius, and $\mathbf{P}_{ij}$ is the relative position vector indicating another robot agent or an obstacle (boundary). Figure 1 shows an example of the danger ratio where a quantized region of collision avoidance is called a cell. The cell direction with the smallest value has the smallest probability of collision and it becomes the next moving direction.

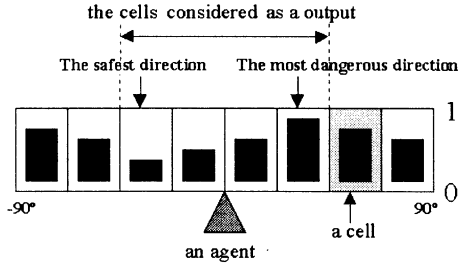


Fig. 1. Example of danger ratio

III. Learning and Evolution of Autonomous Mobile Robots

In this paper, we focus upon supervised learning and evolution with an application to navigation and V-formation of autonomous mobile robots in which the weights are updated through adaptation. The shape of V-formation is shown in Figure 4 and each robot moves and simultaneously maintains constant distance z and angle ϕ with respect to the leader.

(1) compensation of the distance z:

Adjusting the speed to obtain the desired distance with respect to the leader.

$$\Delta s(t)=\begin{cases} -\mathrm{TR} & \text{if } 0\le z\le \mathrm{TR} \\ 0 & \text{if } z-a\le |\mathbf{P}_{ij}^{a}(t)| \le z+a \\ |\mathbf{P}_{ij}^{a}(t)|-z & \text{otherwise} \end{cases} \quad (4)$$

where a is an arbitrary constant, SR and TR is the sensing radius and the threat radius.

(2) compensation of the angle ϕ:

Adjusting the weight w_i^a to keep the desired angle with respect to the leader. The update rule is based on the least-squares method as follows:

$$\mathrm{E}_i(t)=\frac{1}{2}\varepsilon_i(t)^2,\quad \varepsilon_i(t)=(\phi-\theta_{il}(t)) \quad (5\text{-}1)$$

$$w_i^a(t+1)=w_i^a(t)-\alpha\cdot\varepsilon_i(t) \quad (5\text{-}2)$$

where θ_{il} is the angle difference between itself and leader.

(3) strategy of the reinforcement signal:

Adjusting the reinforcement signal for each robot to follow the leader by giving reward or penalty for the current state. The reinforcement signal $\mathrm{r}_i(t)$ is evaluated as follows:

$$\mathrm{r}_i(t)=\begin{cases} 1 & \text{if } \Delta s(t)<0 \\ 0 & \text{if } \Delta s(t)=0 \\ -1 & \text{if } \Delta s(t)>0 \end{cases} \quad (6\text{-}1)$$

$$w_i^a(t+1)=w_i^a(t)-\alpha\cdot r_i(t) \quad (6\text{-}2)$$

Therefore, from (5) and (6), the final update rule is summarized as

$$w_i^a(t+1)=w_i^a(t)-\alpha(\mathrm{r}_i(t)+\lambda\varepsilon_i(t)) \quad (7)$$

where α is a learning coefficient, and λ is a proportional constant. Figure 2 represents the block diagram of the neural network learning structure for each robot agent.

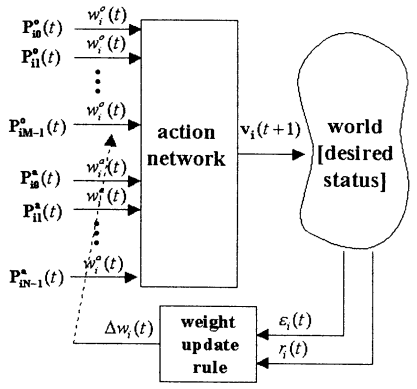


Fig. 2. Behavioral learning structure of an AMR

IV. Simulation Results

1. V-Formation with Flocking and Obstacle Avoidance

From the simulations based on the behavioral learning algorithm, we could obtain various results which are coincident with our strategies. In Figure 3, six autonomous mobile

robots move subsequently around in a flock while avoiding obstacles and boundaries. In the simulations, the desired distance is z=40, the desired angle is ϕ=30°

2. Performance Index

There exists the optimal weight w_i^* which keeps each robot's movement with respect to the leader. This value is optimal in a sense of the desired angle and the desired distance but it may not be optimal in different situations. The optimal weight can be predetermined as follows:

$$w_i^* = \frac{1}{z \cdot \tan\phi}, \quad z > \mathrm{TR} \tag{8}$$

where TR is the threat radius. Therefore, if all the desired conditions are satisfied, the weights of every mobile robot except for the leader will converge to the optimal weight w_i^*, and it is possible to confirm the performance index by investigating the weight trajectories. Figure 4 shows, for six mobile robots, the time evolution of each robot's weight trajectory w_i^a changing while the desired distance is 40 and the desired angle is 25° In Figure 4, it is known that the robots move in a flock at time 550 and thereafter.

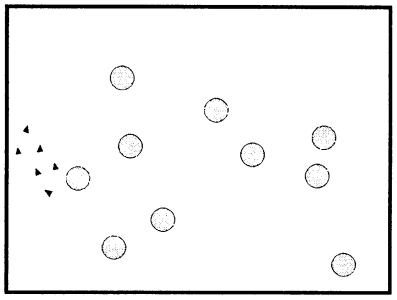


Fig. 3. V-Formation of AMRs

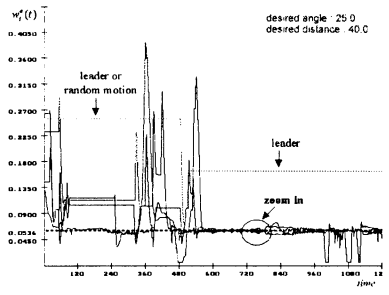


Fig. 4. Weight trajectories of AMRs

V. Conclusions

It is shown that global group behaviors of collective intelligence are induced as results of local interactions between agents and the environment. The suggested control structure of autonomous mobile robots is a combined scheme of the neural network learning and the reinforcement signal which reveals complex behaviors of adaptive systems in the dynamic environment. Especially, the proposed neural network architecture is of simple and distributed type, and from the behaviors of autonomous mobile robot agent, it is recognized that each robot learns and selects the optimal weight adaptively by itself for other robot agents and obstacles.

VI. Reference

[1] C. Langton, *"Artificial Life" in Artificial Life*, C. Langton (ed.), Addison-Wesley, pp.1-47, 1989.

[2] R. A. Brooks, "Behavior Humanoid Robotics", *Proc. of Int. Conf. on IROS*, pp.1-8, 1996

[3] M. J. Mataric, "Designing Emergent Behaviors: From Local Interactions to Collective Intelligence", *Proc. 2nd Int. Conf. on Simulation of Adaptive Behavior*, pp.432-441, 1993.

[4] C. R. Kube and H. Zhang, "Collective Robotic Intelligence", *Proc. 2nd Int. Conf. on Simulation of Adaptive Behavior*, pp.460-468, 1993.

[5] Craig W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model", *Proc. of SIGGRAPH* ,pp.25-34, 1987

[6] M. T. Hagan and H. B. Demuth, Mark Beale, Neural Network Design, PWS Publishing, 1996

[7] J. S. R. Jang, C. T. Sun, and E. Mizutani, "Learning from Reinforcement", *Neuro-Fuzzy and Soft Computing*, pp.258-300, Prentice Hall, 1997

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Cellular Automata Based Neural Networks (CABANN) for Optimal Path Planning

Yong-Goon Jo and Hoon Kang

Dept. of Control and Instrumentation Eng., Chung-Ang Univ.
211, Heuksuk-Dong, Dongjak-Goo, Seoul 156-756, Korea
Tel:+82-2-820-5320 Fax:+82-2-816-1856 E-mail:hkang@cau.ac.kr

Abstract: This paper deals with a simple neural network structure of cellular automata which performs optimal path planning for autonomous mobile robot navigation. Primarily, the structure of cellular automata is well known for the properties of complex behaviors at the edge of chaos and distributed features. We address that a slight modification of each cell in 2D cellular automata into a winner-take-all mechanism of neural networks could achieve optimality in mobile robot path planning. The cell state of the cellular automata based neural networks (CABANN) contains direction and distance information which propagates through the grid-connected lattice structure. The CABANN is applied to optimal path planning of autonomous mobile robots and maze problems.

Key words: cellular automata, optimal path planning, neural networks, autonomous mobile robots

1. Introduction

The optimal path planning is important for a mobile robot to make one's decision autonomously. Recently, parallel process structure and self-organization, self-reproduction and evolution through signal propagation mechanism of cellular automata(CA), are applied to a variety of optimal problems.

CA was introduced in von Neumann's self-organized system and automata theory(von Neumann[1]). He considered that CA consists of a local rule of logic which updates the state of a cell depending on the state of the cell and the neighboring cells. CA has the properties of complex form and behaviors by the interaction between the cell and its neighbors, especially, at the edge of chaos and distributed features(Langton[2]).

In this paper, we address that optimal path planning of autonomous mobile robot can be achieved efficiently and rapidly with the cellular automata based neural networks (CABANN). CABANN has a simple rule obtained from a slight modification of each cell (the states of cell contains direction and distance information which propagates through the grid-connected lattice structure) in 2D CA into a winner-take-all mechanism(Hagan et al.[3], Jang et al.[4]) of neural networks without probability selection through the simulations, and CABANN is applied to the maze problem.

2. Cellular Automata

CA is a discrete dynamical system. Space, time and the state of the system are discrete. Each cell in CA refers to its states and the states of its neighborhood cells and updates its state in synchronization with discrete time by the local interactive rules. These rules are determined by the number of states of the cell and that of its neighboring cells. The structures of CA are categorized into three parts generally - 1-dimension of a belt, 2-dimension of a grid, and 3-dimension of a cube. The diagram of updating the states of the cell looking its states and the states of its neighboring cells is shown in Figure 1(Wolfram[5], Min[6]). The number of possible local interactive rules increases radically in proportion to the number of the cell's states and neighboring cells. In the case of 2-state 1D CA, a neighborhood template including a cell and its immediate neighbors to the left and right, can make 8 possible neighborhood states(000～111) and the state of the cell itself will be mapped to one

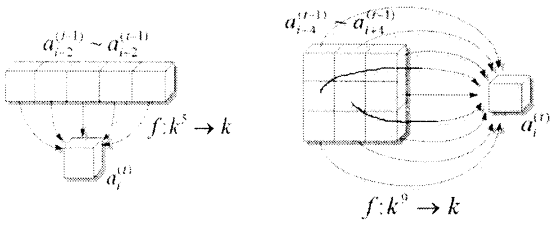


Fig. 1. The process of updating the states looking the states of the cell and its neighboring cells for 1D and 2D CA, respectively.

of the two states. Therefore, there exist $2^{2^3}=256$ possible local rules. For a 4-state 2D CA with von Neumann neighborhood,(von Neumann[1]) there 4^5 possible neighborhood states and $4^{4^5}=4^{1K}$ local rules are possible.

3. An Autonomous Mobile Robot

An autonomous mobile robot(AMR) carries out the given job by itself. In this paper, the AMR has the object to plan the shortest path from the start to the goal, and is considered on the physical hypotheses as follows(Jae-Kal[7-8]).

3.1 Physical Hypotheses for the AMR

① The speed is constant.
② The rotation angle is restricted to 45° .
③ The robot has its own relative coordinates.
④ The sensing radius is fixed.
⑤ The robot measures the distance from itself to all the obstacles and the other AMRs in sensing radius.

3.2 Creating a Road Map

Under the above hypotheses, the AMR creates a road map avoiding the obstacles in given area.

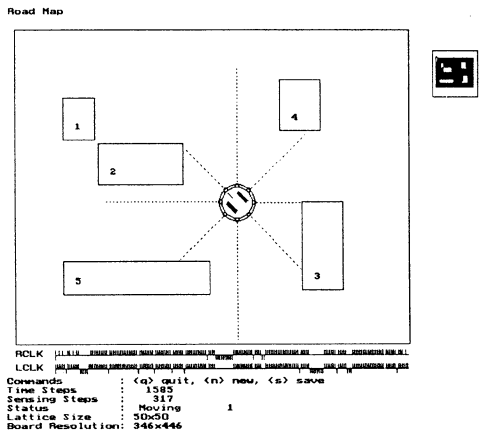


Fig. 2. The AMR is creating a road map moving around the area

The road map is in the form of 2D CA and each cell contains the information on the CA environment whether it is obstacle or not. Figure 2 shows that a mobile robot with 8 sensors and 2 wheels creates a road map moving around the area where the obstacles are placed. The road map relative to the position of the robot is shown in the small box at the upper right corner.

4. CABANN for Optimal Path Planning

In this paper, cellular automata based neural networks(CABANN) finds the optimal path from the start to the goal by applying the winner-take-all mechanism of neural networks to each cell of a 2D CA road map which the AMR has created previously.

4.1 Data Structure of a Cell

The data structure of each cell of a road map is shown in Figure 3 and takes total 32 bits.

① TYPE : a type that each cell of a road map will get, and become one among CLEAR, OBSTACLE, START, and GOAL
② SIGNAL : the signal propagates through each cell from the start until it arrives at the goal, then CABANN obtains the optimal path by going backward to the start along the DIRECTION.
③ DIRECTION : the direction to a neighboring cell contains the shortest distance from the start.
④ DISTANCE : the distance in which SIGNAL propagates from the start, is chosen to be the shortest propagated distance among the neighboring cells containing SIGNAL applying the winner-take-all mechanism.

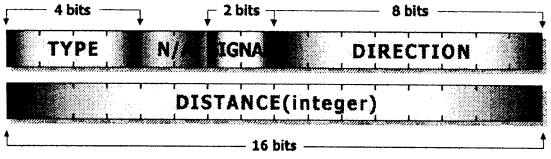


Fig. 3. The data structure of a cell

4.2 An Algorithm for Optimal Path Planning

In a 2D CA road map, every cell excluding the obstacles is considered as a neuron. Each cell shows a basic clue that the optimal path can be acquired according to the local rule(see Figure 4) including the winner-take-all mechanism. Once the start and the goal are given, a signal(START_SIGNAL) propagates from the start to the neighbor cells and the process

goes on. Every cell except the obstacles checks if the signal comes from the top-left cell. If not, it skips over the cell in that direction, and does the same process for the next neighbor cell. Otherwise, it decides that the signal comes from a diagonal or an orthogonal cell. In the case of the diagonal direction, if at least one of two cells adjacent between both the diagonal cell and itself is an obstacle, it skips over this diagonal cell in order to avoid the collision with the obstacles in advance that can be caused by the AMR's physical body. 5 is chosen for the orthogonal direction and $7(\simeq 5\sqrt{2} = 7.071)$ is chosen for the diagonal displacement value.

The displacement value is added to the DISTANCE of a cell in that direction, then compared with the DISTANCE of itself which is initialized as very large number (in this paper, 65535) except that of the start (initialized as 0). By comparing the DISTANCE value in the neighborhood with the DISTANCE value itself, if the latter is larger, it implies that the distance propagated from that directional neighboring cell is shorter than any other neighbor scanned so far. And the cell copies the signal, START_SIGNAL, from the neighbor, updates its DISTANCE as the sum of the DISTANCE of the neighbor and the displacement value, and store the direction to the neighbor as the DIRECTION, i.e. the distance and direction of the winner(shortest) neighborhood cell are chosen. Otherwise, the distance coming from the neighbor is not the shortest. The rule is applied to the other cells except the obstacles scanning from the top-left to the bottom-right successively until the signal reaches the end. We can obtain the optimal path by tracing the DIRECTION backward from the end to the start.

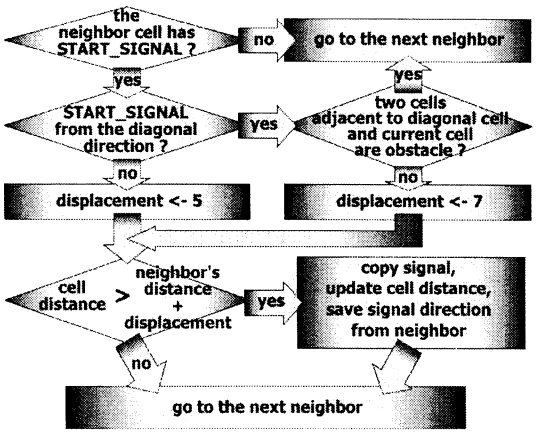


Fig. 4. The flowchart of the algorithm (rule) of a cell for obtaining the AMR's optimal path.

Figure 5 shows that the signal is propagated through the cells from the start, S, to the goal, G, according to the rule for optimal path planning step by step. After the signal reaches the goal, the optimal path is appeared as black arrows. White arrows indicate the shortest direction from the start which is optimal.

5. Simulations

We have applied CABANN to the AMR's optimal path planning and the maze problem. The simulation results show that CABANN can produce reliable and efficient results.

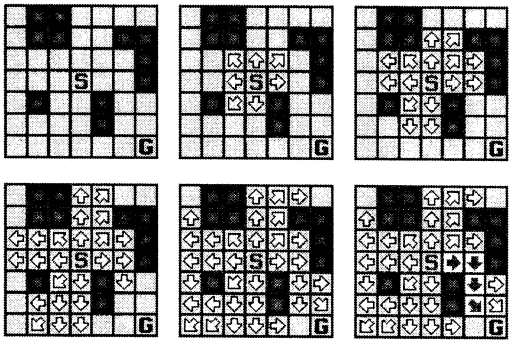


Fig. 5. The process of signal propagation and finding the optimal path

5.1 The AMR's Optimal Path Planning

The AMR used in the simulation satisfies the hypotheses described in section 3.1 and has 8 sensors and 2 wheels. It searches for the area with sufficient time to create a road map in detail where the start and the goal specified manually. The resolution of a road map, i.e. the size of 2D CA is 50×50. The start position is located at the top-left corner and the goal, in the bottom-right. The result is shown in Figure 6. It takes considerably short time of 39 steps for CABANN to propagate the signal from the start to the goal completely. Figure 6 represents the optimal path found by applying the proposed algorithm.

5.2 Maze Problem

Complicated 32×32 mazes used in simulation are shown in Figures 7 and 8. After 98 steps, CABANN could find the optimal path by solving the maze(see Figure 7). We have modified the maze by deleting one wall(obstacle) as shown in the white

circle in Figure 8, and simulated it. At this time, CABANN took 68 steps to find the optimal path, which is shorter than it was before. In fact, CABANN can be also used for adaptive optimal path planning in which the environment changes dynamically.

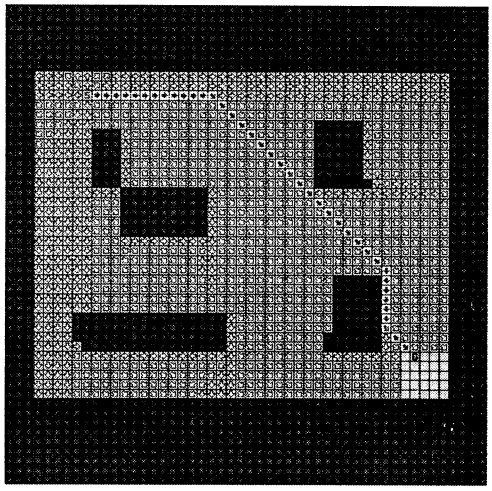


Fig. 6. The optimal path for the AMR found by CABANN (after 39 steps)

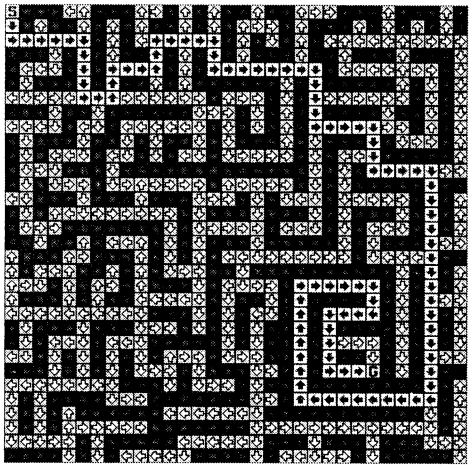


Fig. 7. Solving the maze(after 98 steps)

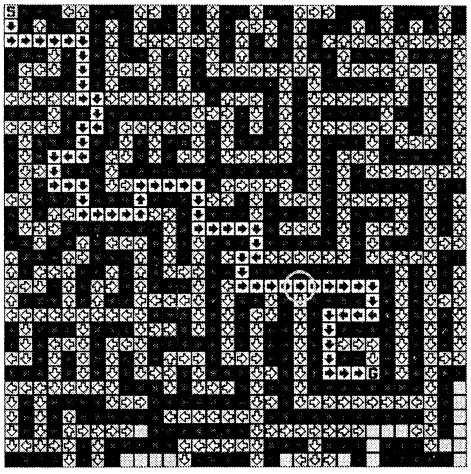


Fig. 8. Deleting one obstacle(wall) in the circle and solving the maze(after 68 steps)

6. Conclusions

We have shown that CABANN proposed in this paper can obtain the optimal path planning for the AMR and solve the complicated maze problem efficiently and rapidly using the simple local rule the winner-take-all mechanism applied to. Also, it deals with the change of environment actively to establish a new optimal path planning as shown in the maze problem simulation. Our further objective is to resolve the problems with the same start and several goals such as TSP(Traveling Salesperson Problem). And we expect refined results if the probabilistic evolutionary selection is added to the algorithm.

7. Reference

1.von Neumann J(1966), The Theory of Self-Reproducing Automata, A. Burks(ed.) A. W. Univ. of Illinois Press, Urbana, 1966

2.Langton C(1992), "Life at the edge of chaos", Artificial Life II, C. Langton (ed.), pp.41-91

3.Hagan MT, Demuth HB, Beale M(1996), Neural Network Design, PWS Publishing Company

4.Jang JSR, Sun CT, Mizutani E(1997), Neuro-Fuzzy and Soft Computing - A Computational Approach to Learning and Machine Intelligence, Prentice-Hall Intl., Inc.

5.Wolfram S(1986), Theory and Application of Cellular Automata, World Scientific Press

6.Min SK, Jae-Kal U, Kang H(1997), Cellular Automata Simulator : CAU-CAS(in Korean), Proceedings of the KIEE '97 Annual Summer Conference, Korea, Jul 21-23, 1997, pp.2296-2298

7.Jae-Kal U, Min SK, Kang H(1997), A Predictive System of Chaotic Behaviors in Collective Autonomous Mobile Robot Agents using Fuzzy Cellular Automata, Proceedings of the 2nd Asian Control Conference. Volume II, Seoul, Korea, Jul. 22-25, 1997, pp.631-634

8.Jae-Kal U, Jo YG, Kang H(1997), Simulation of Collective Autonomous Mobile Robots using Fuzzy Systems(in Korean), Proceedings of KFIS Fall Conference, Suwon, Korea, Nov 29, 1997, pp.445-448

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Evolving Cellular Automata Neural Systems 2

Dong-Wook Lee and Kwee-Bo Sim

Robotics and Intelligent Information system Lab.
School of Electrical and Electronic Eng. Chung-Ang Univ.
221, Huksuk-Dong, Dongjak-Ku, Seoul 156-756, Korea
Tel : +82-2-820-5319, Fax : +82-2-817-0553, http://rics.cie.cau.ac.kr
E-mail : dwlee@cau.ac.kr, kbsim@cau.ac.kr

Abstract

To develope more huge and complex information processing system like human brain, we proposed a new method of constructing artificial brain using genetic algorithms with developmental process. We already developed CA and GA based neural systems that are called "Evolving Cellular Automata Neural Systems 1(ECANS 1)". In ECANS 1, the rule of CA is fixed, and only arrangement of initial cells is evolved by GAs. Because it is limited by the fixed CA rule, the representation space of ECANS1 is very small. So, very simple problems can be applied to. In order to get the sufficient representation space of the network, we propose ECANS 2 in this paper. In ECANS 2, the rule of CA are the objective of the evolution. This method is an algorithm that is based on the characteristics of the biological DNA. In this paper we propose an adequate DNA coding method for evolution of generation rule of cellular automata. In order to verify the effectiveness of our scheme, we apply it to a navigation problem of autonomous mobile robot.

Keywords : Artificial brain, Cellular automata, Neural system, Evolutionary algorithms, DNA coding,

1. Introduction

In recent years, it has been reported that a preliminary result to develope artificial brain at artificial life, soft computing, and robotics related workshop or conference[1]. Most of these researches are the idea from a biological brain model. Specially, the method to evolve the composition rule of neural network is paid attention to. This idea comes from development and evolution of natural living things. The advantage of this method is that big sized network can be made by simple composition rule.

Representative models of development are Cellular Automata(CA), L-System, and biomorph model etc., and evolution models are Genetic Algorithms(GA), Evolution Strategies(ES), Genetic Programming(GP), Evolutionary Programming(EP), and Co-Evolutionary Algorithms(CEA). Boers[2], Gruau[3] etc. proposed the design method of neural networks based on L-system and GA. Garis[4] has studied CA and GP based artificial brain, Sugisaka[1] also has developed artificial brain.

To develope more huge and complex information processing system like human brain, we also use the method of development and evolution. In this paper we propose a new method of constructing artificial brain using genetic algorithms(include DNA coding method) with developmental process. In conventional genetic algorithms, we assume that the arrangement of genotype indicates only the structure of phenotype. In this case, if phenotype is very complex structure then the length of genotype becomes very long. Therefore, it is impossible to evolve complex phenotype. In our system, the translation of genotype to phenotype is developmental process that is cellular automata. The chromosome of GA is a production rule of CA. In general, as the number of states of CA increases, the size of rule table also increases exponentially. So, it is impossible to evolve the rule table. In order to overcome this defect, we propose new encoding method based on biological DNA encoding. This method is an algorithm that is embodied by the characteristics of DNA. DNA coding method[5,6] has much advantages. For example it is suitable to represent the rule, and has good performance when the chromosome is long.

A cell, that is a neuron of neural networks, is modeled on chaotic neuron with firing or rest state like biological neuron. A final output of network is measured by frequency of firing state. The effectiveness of the proposed scheme is verified by applying it to a navigation problem of a robot.

2. Structure of Evolving Cellular Automata Neural Systems

2.1 Neuron Model

The equations of Nagumo-Sato's chaotic neuron model are as follows;

$$y(t+1) = u(x(t+1)) \tag{1}$$

$$x(t+1) = S(t) - \alpha \sum_{d=0}^{t} k^d y(t-d) - \theta \tag{2}$$

where $y(t)$ is a output at time t, $x(t)$ is an inertia state at time t, $S(t)$ is a input at time t, $u(x)$ is a unit step function, k^d is a damping factor of refractoriness having values between 0 and 1, the constant α is a positive parameter, and θ is a threshold of neuron.

The state of a neuron shows chaotic behavior according to the input value. In a chaotic neuron

model, input/output signal is pulse type(Fig. 1), because activation function is a type of unit step function($u(x)$). Therefore, the strength of signal can be obtained by the pulse density modulation.

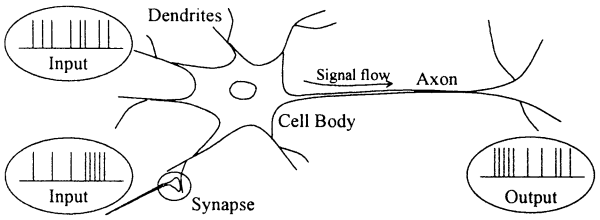


Fig. 1. Biological neuron model with pulse density modulation

2.2 Characteristics of ECANS

Evolutionary concept is applied to constructing the artificial neural network. In practice, the architecture of artificial brain is constructed by translation of chromosome. Because the chromosome is not a blue print of artificial brain but a generation rule, developmental process is necessary to make one. Fig. 2 shows the developmental process of ECANS[7].

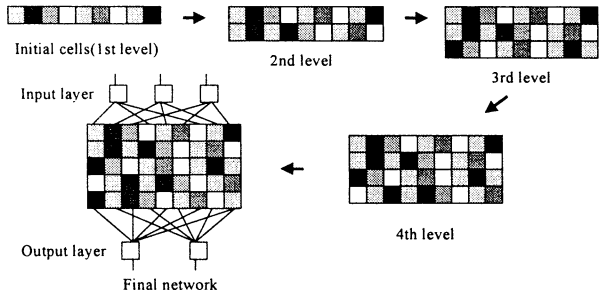


Fig. 2. Developmental process of ECANS

In this paper, basically we use CA model for development. So, the production rule is encoded for a chromosome by DNA coding method, and artificial brain(NNs) emerges from initial cells and interaction of the rules. The state of CA is a cell which have various type of connection. Possible states for use are $27(=3^3)$, 3 direction(right, left, down) and 3 connection(excitation, inhibition, no connection) is possible).

Cellular automata neural system has the following characteristics.

(1) Development of ANN follows CA rule
(2) Each neuron is connected with only its neighbor neuron.
(3) The characteristic equation of a neuron is chaotic neuron model.
(4) The characteristics of the network is determined by not the connection strength(weight) but the connection type.
(5) The strength of the output signal is measured by pulse density.

Table 1 is the difference between ECANS1 and ECANS2, In ECANS1, the object of evolution is the pattern of initial cells, but in ECANS2, the object is the CA rule. As a result, the representation space of ECANS2 is more wide than that of ECANS1.

Table 1. Comparison of ECANS1 and ECANS2

	ECANS1	ECANS2
Initial Cells	Evolution	Fixed
Genotype	Bit string	-
CA Rule	Fixed	Evolution
Genotype	-	DNA code
Representation Space	Middle	Big

3. DNA Coding Method

3.1 Biological DNA

Natural living things have their own DNA[8]. DNA is a genetic code that emerges to the characteristics of individual. Biological DNA consists of nucleotides which have Adenine(A) Thymine(T : Urasil(U) in RNA) Guanine(G), Cytosine(C). A messenger RNA is first synthesized from DNA. Three successive bases called codons are allocated sequentially in the mRNA. These codons are the codes for amino acids. 64 kinds of codons correspond to 20 kinds of amino acid(Table 2). The allocation of amino acid make proteins and proteins make up cells. Translation of mRNA starts on AUG, and comes to an end on UGA.

Table 2. RNA(DNA) codon and amino acid which is generated

	U		C		A		G		
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U
	UUC		UCC		UAC		UGC		C
	UUA	Leu	UCA		UAA	stop	UGA	stop	A
	UUG		UCG		UAG		UGG	Trp	G
C	CUU		CCU	Pro	CAU	His	CGU	Arg	U
	CUC		CCC		CAC		CGC		C
	CUA		CCA		CAA	Gln	CGA		A
	CUG		CCG		CAG		CGG		G
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser	U
	AUC		ACC		AAC		AGC		C
	AUA		ACA		AAA	Lys	AGA	Arg	A
	AUG	Met	ACG		AAG		AGG		G
G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly	U
	GUC		GCC		GAC		GGC		C
	GUA		GCA		GAA	Glu	GGA		A
	GUG		GCG		GAG		GGG		G

* T is used instead of U in DNA

Fig. 3 shows the general structure of genetic code. General point of view, genes are composed of regulatory region and coding region. When the cell state is satisfied to stimulate the regulatory region, the coding region is translated to protein by mRNA. In development model, the representation method of rule is similar to this mechanism. In CA rule, regulatory region is corresponding to a neighbor state of the cell and coding region is corresponding to the next state of the cell.

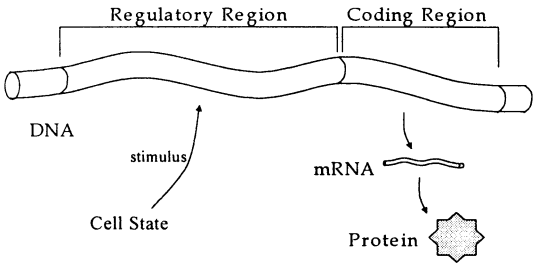


Fig. 3. General structure of genetic code

3.2 Characteristics of DNA Coding Method

Yoshikawa[5] and Deaton[9] proposed new coding method modeled from biological DNA. This artificial DNA consists of nucleotide which have four bases, and it is translated by codon which is three nucleotides like biological mechanism. This method has redundancy of chromosome and overlapping of genes, because start codon and end codon are determined at random. No constraint on crossover point is imposed, so the length of the chromosome is variable. Yoshikawa's method is suitable to represent a rule by preparing a decoding table.

The DNA coding method has the following characteristics.

(1) The length of chromosome is variable(floating representation).
(2) The coding is redundant and overlapped.
(3) It is easy to use knowledge representation.
(4) No constraint on crossover point.

Wu[6] proved the effectiveness of the floating representation by schema analysis in GA. DNA coding is one of the floating representation, because it has not a fixed location for a special meaning. Floating representation has the following features.

(1) It is very effective when the length of chromosome is long
(2) Diversity of population is high; so it has good parallel search ability and recombination ability.

Above characteristics make it possible to evolve the CA rule in our system. Fig 4 shows an example of the overlapped representation and translation.

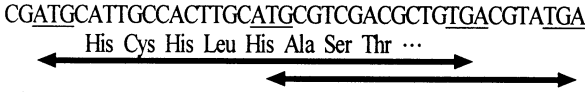


Fig 4. Translation of genetic code / overlapping

3.3 Evolution of CA Rule using DNA Coding Method

In this section, we propose DNA coding method for evolution of CA(Cellular Automata) rule. Eq. (3) shows the general representation of one dimensional CA rule($\Phi_i(\sigma_i)$)[10].

$$\sigma_i^{new} = \Phi_i(\sigma_i) = \phi(\sigma_{i-r}, \cdots, \sigma_{i+r}) \tag{3}$$

where r is the radius of neighborhood.

We consider the case of $r=1$. At this time, according to the state of ith cell and its neighbor states, the next state of ith cell can be represented as a function $(\sigma_{i-1}^j, \sigma_i^j, \sigma_{i+1}^j) \rightarrow \sigma_i^{j+1}$. And if the function ϕ consists of n subfunctions, then ϕ represented in the form

$$\phi = \begin{cases} \phi_1, & \text{condition } 1 \\ \phi_2, & \text{condition } 2 \\ \cdots & \\ \phi_n, & \text{condition } n \end{cases} \tag{4}$$

where the maximum value of n is $(\text{total states})^{2r+1}$.

Condition $1 \sim n$ in eq (4) is determined by its neighborhood, so we use eq. (5) to (7) for condition in this paper.

$$\text{If } \sigma_{i-1}^j = s_1 \text{ and } \sigma_i^j = s_2 \text{ and } \sigma_{i+1}^j = s_3 \tag{5}$$
$$\text{If } \sigma_l = s_1 \text{ and } \sigma_m = s_2 \tag{6}$$
$$\text{If } \sigma_n = s_1 \tag{7}$$

where s_1, s_2, s_3 are the state of cell and $l, m, n \in \{i-1, i, i+1\}$, $l \neq m$.

Eq. (5) that is considered the 3 states, is a specific rule and has highest priority, when the conflict of the rule occurs. Eq. (6)(Eq. (7)) that is considered the 2(1) state(s), is a general rule and has low priority. At this time, subfunction ϕ_k consists of the following operation.

◆ Bit operations used in ϕ_k

AND, OR, XOR, ADD, DIFFERENCE, NOT, 3XOR, State

Table 3. Translation flow of DNA codon

(case 1) 1 state	(case 2) 2 state	(case 3) 3 state
① Start codon		
② Order		
③ State		
④′ A/B (B)	④ A/B (A)	
	⑤ State	
(⑤′⑥″omitted)	⑥′ A/B (B)	⑥ A/B (A)
⑦′ Operation		⑦ State
⑧″ Next State	⑧″ Order	⑧ Next State

"Order" is the order of considering cell of states (③,⑤,⑦) or operation(⑦'). A/B is the abbreviation of And/Break for the control of what states are considered for condition term. Case 1, 2, 3 are that the considering states are 1, 2, 3 in condition term.

Table 4. Translation table of codon

Amino Acid	Phe	Leu	Val	Ser	Pro	Thr	Ala	Tyr	His	Gln	Asn	Lys	Asp	Glu	Cys	Arg	Gly
Order	0	2	1	3	4	1	5	1	3	1	5	4	4	2	5	0	4
State	0	0	1	3	7	2	7	4	5	5	2	5	1	6	3	4	6
A/B	A	A	A	B	A	A	A	A	A	A	A	A	A	A	B	B	B
Operation	0	0	1	3	2	2	1	4	5	5	5	5	6	6	3	4	6

* Ile, Met, Trp and stop codon is used for start codon

Table 4 is the distribution of the condition to DNA codon, we use 8 codon(Ile, Met, Trp, and stop codon) as start codons, so the probability that one nucleotide becomes start codon is 0.125. Translation step of the DNA code in Fig. 4 is as follows;
① ATG : Start codon ② His : Order(i, i-1, i+1) ③ Cys : State(3) ④ His : A/B(A) ⑤ Leu : State(0) ⑥ His : A/B(A) ⑦ Ala : State(7) ⑧ Ser : Next State(3)

The obtained rule is that " $\sigma_{i-1}^{j}=0$ and $\sigma_{i}^{j}=3$ and $\sigma_{i+1}^{j}=7$ then $\sigma_{i}^{j+1}=3$ "

4. Simulation Result

In this paper, ECANS2 is applied to the controller of an autonomous mobile robot. Fig 4 shows the trace of the best individual of 50 generation evolution. Fig 5 shows the input/output data of the robot. For the simulation, we set the conditions and parameters for evolution of ECANS2 as follows;

- Neural Network parameter
  Input neuron : 5(robot has five input sensors)
  Output neuron : 2(output of the two wheel)
  Hidden network size : 10 × 5
- Evolution parameter
  Population : 100
  Length of Initial Chromosome : 512
  Crossover rate : 0.8
  Mutation rate : 0.1

Generation which the best Individual was find : 50
Length of Best Individual : 718

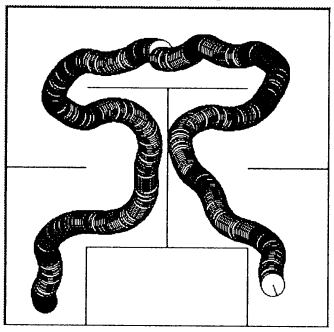


Fig. 4 Trace of Best Individual(Robot)

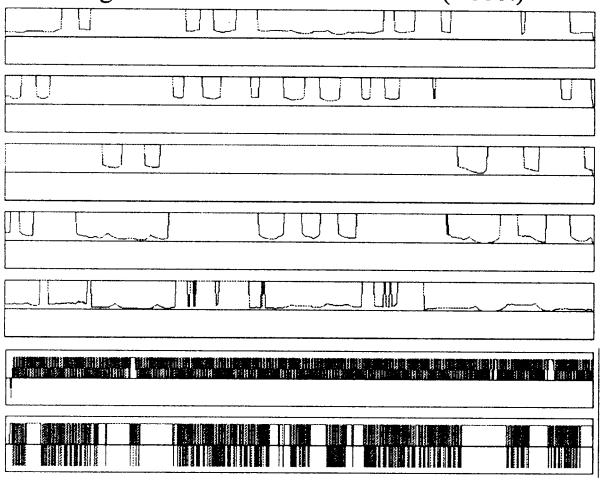


Fig. 5 Input(5) / Output Data(2)

5. Conclusions

We have studied to make artificial brain using developmental and evolutionary concept. In earlier paper(ref. [5]), we proposed ECANS1. ECANS1 has good result in simple problem, however for more complex problem, the ability of the network has limitation. To improve the ability of ECANS1, we change the object of evolution from pattern of initial cells to CA rule. In general, as the total states increases in CA, the number of rules also increases. Therefore we adopt a new type of CA rule that include specific and general rule, and we apply the merit of DNA coding to evolving the CA rule. The proposed method is verified by computer simulation. We are studying to make a artificial brain based on proposed method, and planning to implementation of hardware using FPGA.

Acknowledgement
This work was supported by a grant No. KOSEF 96-01-02-13-01-3 from Korea Science and Engineering Foundation

References

[1] M. Sugisaka, "Design of an artificial brain for robots," Proceedings of The Third International Symposium on Artificial Life and Robotics, pp. (I -2)-(I -11), 1998.
[2] E.J.W. Boers, H. Kuiper, B.L.M. Happel, and S. Kuyper, "Designing Modular Artificial Neural Networks," Proceedings of Computer Science in the Netherlands, pp. 87-96, 1993.
[3] F. Gruau, D. Whitley, "Adding Learning to the Cellular Development of Neural Networks: Evolution and the Baldwin Effect," Evolutionary Computation, vol. 1-3, pp. 213-233, 1993
[4] Hugo de Garis, "CAM-BRAIN : The Genetic Program- ming of an Artificial Brain Which Grows/Evolves at Electronic Speeds in a Cellular Automata Machine," Proceedings of The First International Conference on Evolutionary Computation, vol. 1, pp. 337-339b, 1994.
[5] T. Yomohiro, T. Furuhashi, Y. Uchikawa, "Effect of New Mechanism of Development from Artificial DNA and discovery of Fuzzy Control Rules," Proceedings of IIZUKA '96, pp. 498-501, 1996.
[6] A.S. Wu, R.K. Lindsay, "A Comparison of the Fixed and Floating Building Block Representation in Genetic Algorithm," Evolutionary Computation, vol. 4, no. 2, pp. 169-193, 1996.
[7] D.W. Lee, K.B. Sim, "Evolving Cellular Automata Neural Systems 1(ECANS1)," Proceeding of The 3rd Asian Fuzzy System Symposium, pp. 158-163, 1998. 6.
[8] R.A. Wallace, G.P. Sanders, R.J. Ferl, BIOLOGY : The Science of Life 3rd eds., HarperCollins Publishers Inc., 1991.
[9] Deaton et. al, "A DNA Based Implementation of an Evolutionary Search for Good Encodings for DNA Computation," Proceedings of '97 Magnetics Conference, pp. 267- 271, 1997.
[10] M. Sipper, "Non-Uniform Cellular Automata : Evaluation in Rule Space and Formation of Complex Structures," Artificial Life VI, The MIT Press, pp. 394-399, 1994.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Generalized Asymmetrical BAM

Tae-dok Eom and Ju-Jang Lee
Department of Electrical Engineering
Korea Advanced Institute of Science and Technology
373–1 Kusong-dong Yusong-gu
Taejon 305–701 Korea
email: jjlee@ee.kaist.ac.kr

Abstract

BAM suffers from small memory capacity and abundance of spurious memories although it has the powerful convergence characteristic through the dynamic propagation of noisy data. In this paper, the generalization of usual asymmetrical BAM is proposed to increase the memory capacity and dynamic convergence. The patterns of same Hamming distance can have the different distance measure by adjusting the radii of generalized ellipsoids in hidden layer. In general, it is shown that GABAM is a special form of GRBF(Generalized Radial Basis Function) from the static viewpoint. Output weights and hidden node centers are fixed and unchanged during the learning process. The similarity enables us to utilize the same radius learning techiniques originally designed for GRBF. The property of enhanced noise rejection is investigated by the tests on alphabetical characters.

1 Introduction

In the early 1980's, Hopfield proposed an auto-associative memory model to store and recall information in much the same way as the human brain. In the late 1980's, Kosko extended the auto-associative memory model to a bidirectional one [2]. This two-level nonlinear memory model is based on earlier studies on associative memories [3]. The bidirectional associative memory (BAM) model is more general and powerful than the Hopfield auto-associative memory and includes the Hopfield memory as a special case. A BAM can associate an input pattern with a different stored output pattern of a stored pattern pair, thus allowing bidirectional association. Owing to their good generalization and noise immunity. BAM's are well-suited for pattern recognition.

A BAM consists of neurons arranged in two layers. The neurons in one layer are fully interconnected to the neurons in the other layer. There are no interconnections among neurons in the same layer. The neurons generate action trains which are dependent on the strength of the synaptic interconnections. The instantaneous state of the system is defined by the collective status of each individual neuron (firing or not firing). The memory storage capacity and recall reliability depend on network architecture, as well as the recalling and learning algorithms as well.

The original Kosko BAM suffers from low storage capacity, low recall reliability, and highly spurious memories. Many efforts have been made to improve the performance of the Kosko BAM. Some of these models enhance the Kosko BAM architecture to improve the performance by adding dummy neurons [4], [5], more layers [6], or interconnections among neurons inside each layer [7], while others introduce new learning algorithms to improve the performance [8], [4], [9], [10], [11]. These models all assume logical symmetry of interconnections that the weights from the X-layer to the Y-layer are the same as the weights from the Y-layer to the X-layer. Among those symmetrical models, the symmetrical BAM using the Hamming stability learning algorithm(SBAM) achieves the highest performance[9]. However, the logical symmetry of interconnections not only severely hampers the efficiency of BAM's in pattern storage and recall capability, but also limits their use for knowledge representation and inference[12]. To overcome the drawback of symmetrical interconnections, an asymmetrical BAM model (ABAM) has been proposed[12]. However, the learning algorithm associated with the ABAM model requires linear independence of stored patterns, which limits its storage capacity. All these models can not store more pattern pairs than the number of neurons in their layer. However, the capacity of feedforward multilayer networks and radial basis function network is greater than the number of network neurons. The general BAM (GBAM) suggested by H. Shi *et al.* [1] uses linear separability condition and increases the capacity slightly greater than the number of neurons in its layer.

In this paper, the input weighting matrix is added to induce the more general distance measure than the

Hamming distance. When the input pattern is adjusted, the distance between the input and the stored pattern is calcualted at the virtual layer, which is said virtual cause it only exists during the learning process and merges into the single transition matrix in recall process. The details of GABAM structure is explained in Section II. In Section III, the similarity between GABAM and GRBF is investigated from the static viewpoint. GABAM can be regarded as GRBF with fixed ouput weight, pre-determined basis function centers, and linear activation function. Therefore, input weighting matrix can be trained using backpropagation algorithm. We present the experimental results on the recognition of alphabetical characters in Section IV.

2 The Structure of GABAM

Let $(x^i, y^i), i = 1, \ldots, p$, be the bipolar pattern pairs stored in the memory and the dimensions of x^i and y^i are n and m respectively. KBAM learns the pairs using transition matrix below.

$$W = \sum_i \frac{y^i x^{iT}}{x^{iT} x^i} \tag{1}$$

If the pattern x is applied to input, each y^i is summed in output layer with the weighting factor proportional to $(x^{iT}x)/(x^{iT}x^i) = 1 - 2d_h(x^i, x)$. W^T is used in backward association.

When the human recognize patterns, there are certain feature points more helpful to make a decision. Considering the usefulness of each pixel information, the diagonal matrix $\Lambda^i = diag(\lambda_1^i, \ldots, \lambda_n^i)$ should be multiplied before the correlation matrix. We suggest the new transition matrix W_f as follows.

$$W_f = \sum_i \frac{y^i x^{iT} \Lambda_f^i}{x^{iT} \Lambda_f^i x^i} \tag{2}$$

W_f is not symmetric so backward association is only possible through asymmetric way.

$$W_b = \sum_i \frac{y^i x^{iT} \Lambda_b^i}{x^{iT} \Lambda_b^i x^i} \tag{3}$$

Define the new distance measure as

$$d_{\Lambda^i}(x^i, x) \doteq 1 - 2\frac{\sum_{x_k \neq x_k^i} \lambda_k^i}{\sum_k \lambda_k^i}. \tag{4}$$

Applying the pattern x to input, each y^i is summed with the weighting factor $1 - 2d_{\Lambda^i}(x^i, x)$.

Inducing Λ^i, the convergence characteric can be greatly improved. For the case of auto-association, if $x^1 = [11-1]$ and $x^2 = [-111]$ are stored in KBAM, the memory is trapped into the spurious memory $[-11-1]$. Simply choosing $\forall \Lambda^i = diag[3, 2, 1]$ can avoid the premature convergence.

3 GABAM and GRBF

The matrix Λ^i may be learned by examining overlapped and unique pixels between the training patterns. The less the number of patterns a pixel belongs to, the bigger λ_k, the related element of including pattern, should be. In the other way, the back propagation algorithm for feedforward multilayer networks is applicable.

The individual weighting, $x^{iT}\Lambda^i x$, is equal to $1-(x-x^i)^T\Lambda^i(x-x^i)/2$. The GRBF network can be expressed as

$$y = \sum_i w^i f((x - c^i)^T \Lambda^i (x - c^i)) \tag{5}$$

where activation function $f(u) = e^{-u}$. Then GABAM is translated as GRBF with $w^i = y^i$, $c^i = x^i$, and $f(u) = 1 - u/2$. BAM has no hidden layer in natural way. However, augmenting weighting matrix Λ^i derives the virtual hidden layer from the transition matirx. It is said virtual cause it only exists during the learning process and merges into the single transition matrix in recall process. If the hyperbolic tangent function replaces the usual sign output activation function, the whole network is differentiable and any kind of gradient decent algorithm can be adopted to findΛ^i's which minimize the overall classification errors.

Learning Process : For the learning process, backpropagation algorithm, which uses only the first derivatives(*i.e.* not including Hessian matrix), is adopted. The energy function is denoted by $E = \|y - y^d\|^2/2 = \sum_k (y_k - y_k^d)^2/2$. If the learning constant η is sufficiently small, this pattern learning has the same performance with the batch learning for all training pattern pairs. First, the feedforward propagation is performed for the current training pairs (x^d, y^d). Calculating the output error, the error is propagated backwards. δ_k^o and δ_i^h denote the backpropagated error in output and hidden layer respectively. Then,

$$\delta_k^o = (y_k - y_k^d) \tag{6}$$

$$\delta_i^h = \sum_k y_k^i (1 - tanh^2(y_k)) \delta_k^o \quad (7)$$

$$\frac{\partial E}{\partial \lambda_j^i} = -2 \frac{m(x_j^i \neq x_j) \sum_k \lambda_k^i - \sum_{x_k^i \neq x_k} \lambda_k^i}{(\sum_k \lambda_k^i)^2} \delta_i^h \quad (8)$$

where

$$m(condition) = \begin{cases} 1, & if\ condition\ is\ true, \\ 0, & otherwise. \end{cases} \quad (9)$$

Finally, λ_j^i is updated by

$$\Delta \lambda_j^i = -\eta \frac{\partial E}{\partial \lambda_j^i}. \quad (10)$$

Recall Process : The recall process is same with the other BAM's. the input pattern continues propagating forwards and backwards repeatedly until it reaches a fixed point.

Hidden Node Addition : Like GRBF, the capacity of GABAM can be enhanced through the node addition techniques. Though the activation function of hidden layer is linear that it suffers from low function approximation capability, additional nodes at centers of malfunction bring about capacity improvement. After some interval, the percentage of correct recalls are calculated and number of nodes inversly proportional to the performance ard added.

4 Experimental Results

The performance of an associative memory is usually measured in terms of its attraction capability presented with the noisy patterns. In this section, we compare the property of GABAM with that of the most promising SBAM, ABAM, and GBAM. The 26 test pattern pairs are the matched small case and large case alphabet letters. The dimensions of input and output patterns are 49 (7 by 7).

Figure 1 shows the recall result of four BAM's in the case of noise existence from 0 to 20 percentage. Is suggensted the three variants of GABAM which use the different number of nodes in virtual layer and training patterns. The first has 26 hidden nodes and trained in the Hamming ball of radius 1. The second adds 26 more hidden nodes after learning the first and trained in the same Hamming ball again. The third also trained 26 additional nodes differently in the radius 2. The first variant results in the intermediate performance between GBAM and ABAM. The graphs of the second and GBAM are almost overlapped over all noisy data set. In fact, the learning process of GBAM stems from the linear separability condition and 49 by 49 weights are trained to place the right hyperplanes. The GABAM updates 49 by 52 weights through backpropogation. In fact, activation functions in virtual layer can not become the functional bases due to its linearity , as is also the reason called virtual. Though the linear separability limits the capacity of both models. GABAM is trained fast and easily starting from the predetermined output weights and hidden nodes centers. The third variant is traind in the broader set and overwhelms every BAM model except for the test on Hamming ball of radius 1.

Figure 2, 3, 4 reflects the tendency of convergence for the three variants. Increased static mapping induces enhanced convergence property. As amount of noise grows, the attraction effect is gradually reduced and corrupted to the generic noise level.

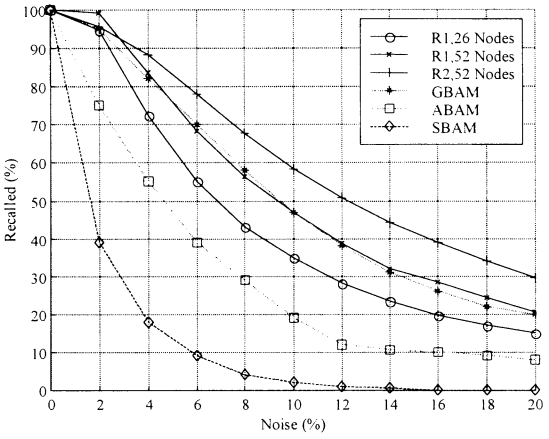


Figure 1: Comparision of Models

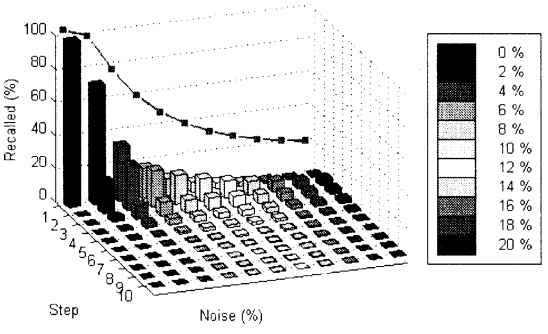


Figure 2: GABAM : R1 and 26 hidden nodes

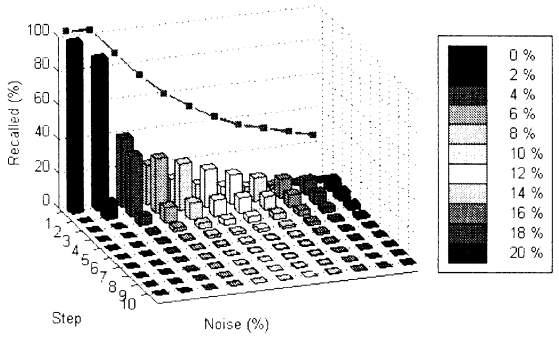


Figure 3: GABAM : R1 and 52 hidden nodes

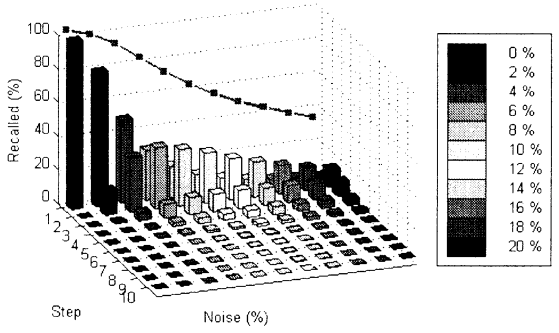


Figure 4: GABAM : R2 and 52 hidden nodes

5 Conclusion

GABAM is proposed and enhances the attraction property inducing the additional input weights. In general, GABAM is shown to be a special case of GRBF. The virtual hidden layer is a fruitful concept which enables GABAM to utilize the various learning techniques. It also merges into the single transition matrix during the recall process. GABAM has the two useful characteristics. One is the enhance discrimination propery for the patterns resident on the border of stored patterns. The other lies on choosing training patterns arbitrarily by supervised learning. Human does not recognize merely dependent on the Hamming distance.

References

[1] H. Shi, Y. Zhao, and X. Zhuang, "A general model for bidirectional associative memories," *IEEE Trans. SMC*, vol. 28, no. 4, Aug, 1998.

[2] B. Kosko, "Adaptive bidirectional associative memories," *IEEE Trans. SMC*, vol. 18, no. 1, pp. 49-60, 1988.

[3] Y. Hirai, "A Model of human associative processor (HASP)," *IEEE Trans. SMC*, vol. 13, no. 5, pp. 851-857, 1983.

[4] Y. Wang, J. Cruz, and J. Mulligan, "Two coding strategies for bidirectional associative memory," *IEEE Trans. Neural Networks*, vol. 1, no. 1, pp. 81-92, 1990.

[5] Y. Wang, J. Cruz, and J. Mulligan, "Guaranteed recall of all training pairs for bidirectional associative memory," *IEEE Trans. Neural Networks*, vol. 2, no. 6, pp. 559-567, 1991.

[6] H. Kang, "Multilayer associative neural network (MANN's) : Storage capacity versus perfect recall," *IEEE trans. Neural Networks*, vol. 5, pp. 812-822, Sept. 1994.

[7] Z. Wang, "A bidirectional associative memory based on optimal linear associative memory," *IEEE Trans. Comput.*, vol. 45, pp. 1171-1179, Oct. 1996

[8] M. Hassoun and A. Youssef, "A high-performance recording algorithm for Hopfield model associative memories," *Opt. Eng.*, vol. 27, no. 1, pp. 46-54, 1989.

[9] X. Zhuang, Y. Huang, and S. Chen, "Better learning for bidirectional associative memory," *Neural Networks*, vol. 6, no. 8, pp. 1131-1146, 1993.

[10] H. Oh and S. Kothari, "Adaptation of the relaxation method for learning in bidirectional associative memory," *IEEE Trans. Neural Networks*, vol. 5, pp. 573-583, July 1994.

[11] C. Wang and H. Don, "An analysis of high-capacity discrete exponential BAM," *IEEE Trans. Neural Networks*, vol. 6, no. 2, pp. 492-496, 1995.

[12] Z. Xu, Y. Leung, and X. He, "Asymmetric bidirectional associative memories," *IEEE Trans. SMC*, vol. 24, pp. 1558-1564, Oct. 1994.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Artificial Immune System for Realization of Cooperative Strategies and Group Behavior in Collective Autonomous Mobile Robots

Dong-Wook Lee, Hyo-Byung Jun, and Kwee-Bo Sim

Robotics and Intelligent Information system Lab.
School of Electrical and Electronic Eng. Chung-Ang Univ.
221, Huksuk-Dong, Dongjak-Ku, Seoul 156-756, Korea
Tel : +82-2-820-5319, Fax : +82-2-817-0553, http://rics.cie.cau.ac.kr
E-mail : dwlee@cau.ac.kr, kbsim@cau.ac.kr

Abstract

In this paper, we propose a method of cooperative control(T-cell modeling) and selection of group behavior strategy(B-cell modeling) based on immune system in distributed autonomous robotic system(DARS). Immune system is living body's self-protection and self-maintenance system. Thus these features can be applied to decision making of optimal swarm behavior in dynamically changing environment. For the purpose of applying immune system to DARS, a robot is regarded as a B cell, each environmental condition as an antigen, a behavior strategy as an antibody and control parameter as a T-cell respectively. The executing process of proposed method is as follows. When the environmental condition changes, a robot selects an appropriate behavior strategy. And its behavior strategy is stimulated and suppressed by other robot using communication. Finally much stimulated strategy is adopted as a swarm behavior strategy. This control scheme is based on clonal selection and idiotopic network hypothesis. And it is used for decision making of optimal swarm strategy. By T-cell modeling, adaptation ability of robot is enhanced in dynamic environments.

Keywords : Distributed Autonomous Robotic System, Group Behavior, Immune System, Modeling of B-cell and T-cell

1. Introduction

The most significant feature of distributed autonomous robotic system(DARS) is that each robot perceive the its environments such as object and the other robot's behavior etc., and they determine the their behavior independently, and cooperate with the other robots in order to perform the given tasks very well[1～4]. Distributed Autonomous Robotic System(DARS) has no function to integrate the whole system. But a robot, that is component of the system, individually understands objective of the system, environment, and behavior of other agents etc., and decides its behavior autonomously to cooperate with other agent and to establish and maintain order of the whole system. Immune system is also distributed autonomous system that protects and maintains living body[5][6]. The components of immune system do not follow commands of the brain but cope with environment autonomously. In this point of view, we analogize DARS from immune system. So this analogy can be used for mechanism that decides group behavior strategy of DARS.

Immune system has various function that are ability to recognize foreign pathogens, ability to process information, ability to learn and memorize, ability to discriminate between self and non-self, and ability to keep up harmony of the whole system. It is thought that these functions of immune system can be applied to various engineering fields[7～13].

In this paper, immune system is applied to making action strategy in collective autonomous mobile robots with dynamic environmental changing. In order to improve the adapting ability of robot, we add T-cell model to the immune network equation.

2. Immune System Modeling

2.1 Immune System

The protection system that eliminates foreign substances that invade living body is called immune system[8]. The basic components of the immune system are lymphocytes that occur as two major types, B cells(B lymphocytes) and T cells(T lymphocytes). B cells take part in humoral immunity that secrete antibody, and T cells take part in cell mediated immunity. Each of B cells has distinct molecular structure and produces 'Y' shaped antibodies from its surfaces. The antibody recognizes antigen that is foreign material and eliminates it. This antigen-antibody relation is innate immune response.

Most antigens have various antigen determinants that is called epitope. In order to grab and latch onto antigen, antibody possesses a structure which is to the epitope as a key is to a lock. This corresponding structure, 'key', is called paratope of the antibody. And recent studies on immunology have clarified that each type of antibody has also its specific antigen determinant called idiotope. So, it is thought that antigen-antibody reaction occurs between antibodies.

2.2 Artificial Immune System(AIS)

Jerne[14] who is an immunologist proposed idiotopic network hypothesis(immune network hypothesis) based on mutually stimulus and suppression between antibodies. Farmer[15] proposed immune network equation of Jerne's hypothesis. In this paper, to improve the adaptation ability of the system, we propose a modified immune network equation which is added helper and suppressor T-cell model. Eq. (1)~(3) are the modified immune network equations that are modeled relationship of antigen, B-cell(antibody), and T-cell of immune system.

$$S_i(t) = S_i(t-1) + \left(\alpha \frac{\sum_{j=1}^{N} (m_{ij} s_j(t))}{N} - \alpha \frac{\sum_{k=1}^{N} (m_{ki} s_k(t))}{N} + \beta g_i - c_i(t-1) - k_i \right) s_i(t) \quad (1)$$

$$s_i(t) = \frac{1}{1 + \exp(0.5 - S_i(t))} \quad (2)$$

$$c_i(t) = \eta (1 - g_i(t)) S_i(t) \quad (3)$$

where $i, j = 0, \cdots, N\text{-}1$, N is total number of antibody types, $S_i(t)$ is stimulus value of antibody i, $s_i(t)$ is concentration of antibody i, $c_i(t)$ is concentration of T-cell which control antibody, γ_{ij} is mutual stimulus coefficient of antibody i and j, g_i is affinity of antibody i and antigen, α, β, η are constants.

In eq. (3), when the stimulus value of antigen($g_i(t)$) is big and the stimulus value of antibody($S_i(t)$) is small, the concentration of T-cell($c_i(t)$) is small. Therefore, in this case c_i(t) take a role of helper T-cell that stimulate B-cell. On the contrary, the stimulus value of antigen is small and the stimulus value of antibody is big, the $c_i(t)$ is big. So, it take part in suppressor T-cell. In biological immune system the helper T-cell activate B-cell when the antigen invade it, and the suppressor T-cell prevent the activation of B-cell when the antigen was eliminated. By adding T-cell modeling, performance of system(making group behavior) is improved.

Table 1. The role of T-cell with different states

$g_i(t)$	$S_i(t)$	$c_i(t)$	state	role of Tcell
big	small	very small	invading	helper Tcell
big	big	small	eliminating	-
small	big	big	eliminated	supressor Tcell
small	small	small	stable	-

The main reason why the T-cell model is added is that the system adapts the environment quickly by recovery of the concentration of antibody to the initial state when the antigens are removed by antibodies. This is more similar to the biological immune system.

3. Group Control Algorithm based on AIS

In this paper we developed the artificial immune network based control algorithm of group in DARS. In general, the process of DARS to execute the give task is (1) Group behavior(for movement of group or arrangement of the system) → (2) Task execution → (3) Detection of environment change → (4) Arbitration of behavior strategy or control of group → (1) Group behavior ···. In order to execute this process, we modeled the group behavior and its relation in sec 3.1 and propose the algorithm for behavior arbitration for group behavior in sec. 3.2.

3.1 Modelling of Group Behavior

Objective of the system and the proposed algorithm

The objective of the system is for robots to find and execute tasks, the tasks are spread out in the environment.

By proposed immune algorithm, strong strategy is selected as a swarm strategy. Namely, all robot select strong strategy. After that, group behavior can be realized. When environmental changes occur, all robot can adapt itself to new situation rapidly. This algorithm is modeled based on clonal selection and immune network hypothesis.

Definition of the antigen

According to the distribution of task, the density of the task is classified into four levels that are ① high, ② medium, ③ low, and ④ nothing. For each of these environments, a robot faces with several strategies that are [1] Aggregation, [2] Random search, [3] Dispersion, and [4] Homing. Accordingly, each environment of four level is regarded as antigens and each of these strategies are regarded as antibodies.

Definition of the antibody

In this paper, objective of the system is to find and execute tasks. So in order to find and execute spread tasks, for each environment that is defined above, four strategies are introduced. Each strategy (antibody) and its definition are as follows.

- Aggregation(Ab_0) : the ability of a group of agents to gather in order to establish and maintain some maximum inter-agent distance.
- Random Search(Ab_1 : basic strategy) : the ability to find task by moving random direction.
- Dispersion(Ab_2): the ability of a group of agent to spread out in order to establish and maintain some minimum inter-agent distance.
- Homing(Ab_3): the ability to find and go a particular region or location.

Stimulus value of antigen to antibody is defined as Fig. 1, according to the percentage of task detection during past given times(*atime*). When density of task is high, stimulus value for

aggregation(Ab_0) is also high. Likewise, other functions are defined. Fig. 1 represents the definition of every function($g_0 \sim g_3$) which has overlapping region.

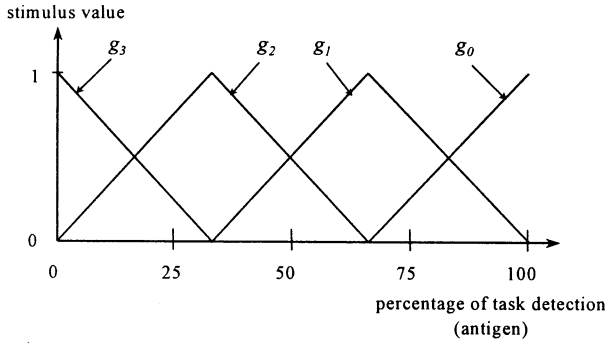


Fig. 1 Stimulus function of antigen to antibody(g_i)

3.2 Decision making of Group

When a robot carries out the given task in DARS, the robot must decide its behavior by local information. At this time, the robot is not able to know all the information of the whole system. Thus it is difficult to realize group behavior such as movement of group or arrangement of system. The idea of the immune response is applicable to arbitration of group strategy in DARS.

Once a robot decides its behavior by perception of its environment, its behavior strategy is stimulated and suppressed by relationship of other robot which encounters. Naturally, this process is accomplished by local communication of autonomous mobile robot. When a robot encounters other robot, if other robot's strategy has the same or similar strategy, this strategy is stimulated, if not, this strategy is suppressed. At this time, if a robot is stimulated very much, it's behavior is regarded as adequate one in the system. So this robot transmits its strategy to other robots. In this way swarm strategy is decided. The algorithm is as follows;

(For each robot)

[Step 1] Initialize stimulus value and concentration of antibody for all action strategies.
$t = 0$
$S_i(0) = s_i(0) = 0.5$ for $i = 0,\cdots,N\text{-}1$
where N is the number of action strategies.

[Step 2] Select and execute strategy(antibody) that has bigger concentration of antibody(s_i) than others.(In start, basic strategy(Ab_1) is selected.)

[Step 3] When a robot encounters other robot, they stimulate and suppress each other using local communication. Stimulus value of B cell(S_i), concentration of antibody(s_i), and concentration of T-cell are calculated by equation (4), (2), and (3) respectively.
The stimulus term and the suppression term are put together as the second term of equation (4), because γ_{ij} is plus(stimulus) or minus(suppression) value.

$$S_i(t) = S_i(t-1) + \left(\alpha \frac{\sum_{j=0}^{N-1} \gamma_{ij} s_j(t-1)}{N} + \beta g_i - c_i(t-1) - k_i \right) s_i(t) \quad (4)$$

where $i, j = 0,\cdots,N\text{-}1$, s_j is concentration of other robot's antibody, γ_{ij} is mutual stimulus coefficient of antibody i and j(table 2), α, β are parameters of response rate of other robots and environment(antigen).

If a robot has a strategy which was stimulated over upper threshold($\bar{\tau}$), then it becomes excellent robot.
→ it can transmit strategy to inferior robot when it encounters inferior robot.
If a robot has all strategy which was stimulated under lower threshold($\underline{\tau}$), then it becomes inferior robot.
→ it receives good strategy from excellent robot when it encounters excellent robot.

$$\bar{\tau}(\text{upper threshold}) = 0.622 \left(= \frac{1}{1+e^{-0.5}}\right) \quad (5)$$

$$\underline{\tau}(\text{lower threshold}) = 0.378 \left(= \frac{1}{1+e^{0.5}}\right) \quad (6)$$

[Step 4] If a inferior robot encounters excellent robot, it receives all strategies and renews concentration of each strategy.

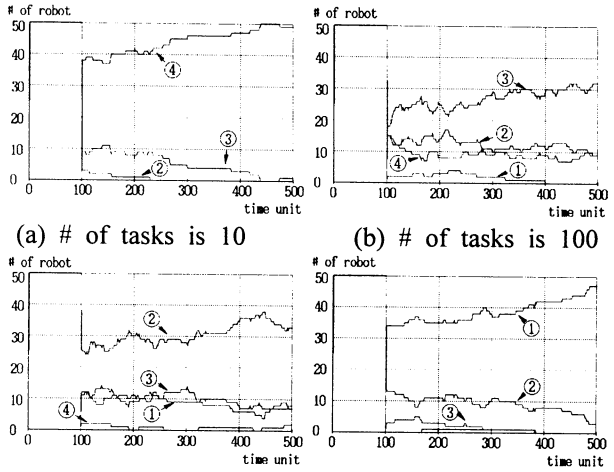
[Step 5] t = t + 1, go to [Step 2].

4. Simulation Result

The simulation environment conditions for verifying the effectiveness of the proposed swarm immune algorithm are set as follows.

- Number of robots : 50
- Working Area : 10m × 10m
- Robot size : 5cm × 5cm
- Objective of the system : find and execute given spread tasks.(In this simulation, robots does not execute given tasks. But we can see how swarm strategy is decided.)
- Communication radius : 75 cm
- Parameter values :
  $\alpha = 0.5$, $\beta = 0.005$, k = 0.002
- Antigen evaluation time(atime) : 100 unit time
  (During 1 unit time, a robot can change its direction or can move 2.5 cm forward.)
- Condition of the system
  (case 1) number of tasks is 10
  (case 2) number of tasks is 100

(case 3) number of tasks is 200
(case 4) number of tasks is 500



(a) # of tasks is 10 (b) # of tasks is 100

(c) # of tasks is 200 (d) # of tasks is 500

Fig. 2 Simulation results with immune algorithm

Fig. 2 is a simulation result in case the proposed swarm immune algorithm is used. Each robot decides its strategy by mutual relationship. In this case, strong strategy is adopted as a swarm strategy. Fig. 2 (a) shows that strategy 4(Ab3) is adopted as a swarm strategy, and the remainder cases in Fig. 6 depict that strategy 3(Ab2), 2(Ab1), 1(Ab0) is selected respectively. Therefore proposed swarm immune algorithm is applied to making swarm strategy in collective autonomous mobile robots with local information. This simulation don't show the advantage of adding the T-cell model, because the environment is not change. We continue to change the simulator, we will acquire the good result soon.

5. Conclusions

In this paper we proposed the algorithm based on immune system to achieve the goal of DARS. We found the analogy between DARS and immune system, and applied to making swarm strategy in DARS. We made a condition that swarm behavior emerged, it could be achieved only when a successful strategy is selected by all robots of the system. This is based on clonal selection that a successful clon is selected and proliferated, and immune network hypothesis that is modeled after mutual action of antibodies. In order to improve the adapting ability of robot in changing environment, we propose the T-cell model of the immune network equation.

Acknowledgement

The Authors wish to acknowledge the financial support the Korea Research Foundation made in the program year of 1997.

References

[1] A. Asama et. al. eds, *Distributed Autonomous Robotic Systems I, II*, Springer-Verlag, 1994, 1996.

[2] D.W. Lee, K.B. Sim, "Behavior Learning and Evolution of Collective Autonomous Mobile Robots using Distributed Genetic Algorithms," *Proc. of 2nd AScian Control Conference*, vol.2, pp. 675-678, 1997. 7

[3] D.W. Lee, K.B. Sim, "Development of Communication System for Cooperative Behavior in Collective Autonomous Mobile Robots," *Proc. of 2nd AScian Control Conference*, vol. 2, pp. 615-618, 1997. 7.

[4] D.W. Lee, K.B. Sim, "Artificial Immune Network-based Cooperative Control in Collective Autonomous Mobile Robots," *Proc. of the 6th IEEE Int. Workshop on RO-MAN*, pp. 58-63, 1997. 9

[5] I. Roitt, J. Brostoff, D. Male, Immunology 4th edition, Mosby, 1996.

[6] R.A. Wallace, G.P. Sanders, R. J. Ferl, BIOLOGY : The Science of Life 3rd eds., HarperCollins Publishers Inc., 1991.

[7] Y. Ishida, N. Adachi, "An Immune Algorithm for Multiagent : Application to Adaptive Noise Neutralization," *Proc. of IROS 96*, pp. 1739-1746, 1996.

[8] S. Forrest, B Javornik, R.E. Smith, A.S. Perelson, "Using Genetic Algorithms to Explore Pattern Recognition in the Immune System," *Evolutionary Computation*, vol. 1, no. 3, pp. 191-211, 1993.

[9] A. Ishiguro, Y. Watanabe, Y. Uchikawa, "An Immunological Approach to Dynamic Behavior Control for Autonomous Mobile Robots," *Proc. of IROS 95*, pp. 495-500, 1995.

[10] A. Ishiguro, Y. Shirai, T. Kendo, Y. Uchikawa, "Immunoid : An Architecture for Behavior Arbitration Based on the Immune Networks," *Proc. of IROS 96*, pp. 1730-1738, 1996.

[11] H. Bersini, F.J. Varela, "The Immune Recruitment Mechanism: A Selective Evolutionary Strategy," *Proc. of 4th Int. Conf. on Genetic Algorithms*, pp. 520-526, 1991.

[12] N. Mitsumoto et al., "Micro Autonomous Robotic System and Biologically Inspired Immune Swarm Strategy as a Multi Agent Robotic System," *Proc. of Int. Conf. on Robotics and Automation*, pp. 2187-2192, 1995.

[13] P. D'haeseleer, S. Forrest, P. Helman, "An Immunological Approach to Change Detection : Algorithms, Analysis and Implications," *Proc. of IEEE Symp. on Security and Privacy*, 1996.

[14] N.K. Jerne, "Idiotopic Network and Other Preconceived Ideas," *Immunological Rev.*, vol. 79, pp. 5-24, 1984.

[15] J.D. Farmer, N.H. Packard, and A.S. Perelson, "The immune system, adaptation, and machine learning," *Physica 22-D*, pp. 184-204, 1986.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Genetic Programming-Based Alife Techniques for Evolving Collective Robotic Intelligence

Dong-Yeon Cho
Artificial Intelligence Lab (SCAI)
Dept. of Computer Engineering
Seoul National University
Seoul 151-742, Korea
Phone: +82-2-880-7302
dycho@scai.snu.ac.kr

Byoung-Tak Zhang
Artificial Intelligence Lab (SCAI)
Dept. of Computer Engineering
Seoul National University
Seoul 151-742, Korea
Phone: +82-2-880-1833
btzhang@scai.snu.ac.kr

Abstract

Control strategies for a multiple robot system should be adaptive and decentralized like those of social insects. To evolve this kind of control programs, we use genetic programming (GP). However, conventional GP methods are difficult to evolve complex coordinated behaviors and not powerful enough to solve the class of problems which require some emergent behaviors to be achieved in sequence. In a previous work, we presented a novel method called fitness switching. Here we extend the fitness switching method by introducing the concept of active data selection to further accelerate evolution speed of GP. Experimental results are reported on a table transport problem in which multiple autonomous mobile robots should cooperate to transport a large and heavy table.

Keywords: genetic programming, artificial life, multiagent learning, fitness switching, training data selection

1 Introduction

Many types of animals seek the company of others. There are several benefits to be gained by gathering into groups with others of the same species, including safety from predators, access to mates, and help in finding food [1].

Similarly, multiple robotic agents can perform some tasks more easily or effectively by cooperation with others. Most of these tasks, however, are so complex that human programming alone is not effective enough to take into account all the detail of real-world situations. In this case, genetic programming provides a way to search for the most fit computer program that solves the problem given by training examples [2]. GP starts with an initial population of randomly generated computer programs. The computer programs are usually represented as trees composed of function and terminal symbols appropriate to the problem domain. They are evolved to better programs using the selection and genetic operations. The ability of the program to solve the problem is measured as its fitness value.

Due to this powerful expressiveness, a number of attempts have been made to evolve cooperative behavior of a group of simple robotic agents using genetic programming. Koza and Bennett [2, 3] used genetic programming to evolve a common program that causes foraging of foods by an ant colony. Luke et al. [4] explored various strategies to develop cooperation in predator-prey environments. Iba [5] studied three different breeding strategies (homogeneous, heterogeneous, and coevolutionary) for cooperative robot navigation. Luke [6] used GP to evolve soccer behaviors of software robots in the RoboCup Soccer Server.

However conventional GP methods are difficult to solve the problems which require some emergent behaviors to be achieved in sequence. Moreover, it requires enormous computational time to evolve complex programs. For these reasons, we used a novel method called fitness switching presented in a previous work [7] and active data selection introduced by one of the authors for efficient training of neural networks [8].

The paper is organized as follows. Section 2 describes the task. Section 3 presents the general framework for fitness switching. Section 4 describes the GP approach with active data selection. Section 5 shows experimental results of the presented methods. Section 6 discusses the result and suggests further work.

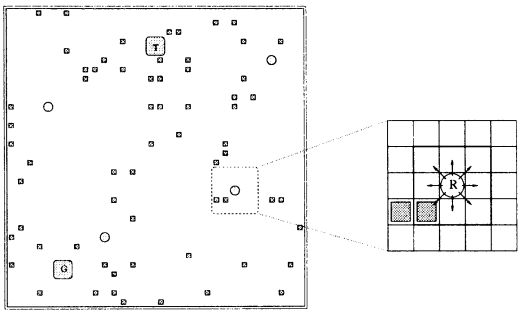


Figure 1: The environment for multiagent learning.

2 The Table Transport Problem

The table transport problem we consider in this paper is defined as follows. In an $n \times n$ grid world, a single table and four robotic agents are placed at random positions, as shown in Figure 1. In addition, a specific location is designated as the destination. A fixed number of obstacles are also placed in the grid. The goal of the robots is to transport the table to the destination in group motion. The robots need to move in herd since the table is too heavy and large to be transported by single robots.

Each robot i ($i = 1, .., N_{robots}$) is equipped with a same control program A_i. The robots activate A_i's in parallel to run a team trial. At the beginning of the trial, the robot locations are chosen at random in the arena. They have different positions and orientations. During a trial, each robot has a repertoire of actions. It can move forward in the current direction (N, E, S, W, NE, SE, SW, NW) or remain on the current position. The direction of the movement can be chosen randomly to avoid collision with obstacles or other robots. The robots have a limited visual field of range 1 to each movable direction.

At the end of the trial, each robot i gets a fitness value which was measured by summing the contributions from various factors. The goal of genetic programming is to find control programs leading to efficiently transporting the table from the initial position to the goal position.

3 Evolving Cooperative Behaviors by Fitness Switching

The problem described in Section 2 requires two different behaviors homing and herding, that is, the robots need first to get together around the table and then transport it in team to the destination. In fitness switching method, different parts of a genetic tree are responsible for different behaviors and for each of the subtrees a basis fitness function is defined. Thus, our genetic program tree can be considered as a composition of two subtrees S_1, S_2, where S_1 is responsible for homing and S_2 is for herding.

In the experiments we have used the following fitness functions for S_1 and S_2:

$$f_1 = \sum_{r=1}^{4}\{c_1 \max(X_r, Y_r) + c_2 S_r + c_3 C_r - c_4 M_r + K\} \quad (1)$$

$$f_2 = \sum_{r=1}^{4}\{c_1 \max(X_r, Y_r) + c_2 S_r + c_3 C_r - c_4 M_r + c_5 A_r + K\} \quad (2)$$

The definitions of the symbols used in above equations are provided in Table 1. The target position for homing is the initial position of the table while the target position for herding is the destination of the table.

Fitness of programs is measured at each generation as follows:

1. Measure the fitness of the left *subtree* by f_1.
2. Measure the fitness of the right *subtree* by f_2.
3. The fitness of the program is defined as $F = f_1 + f_2$.

The advantage of this method is the ability of concurrent evolution of primitive cooperative behaviors and their coordination. The following procedure gives more detailed description of this method.

```
For g = 1 to MAX_GEN
    For all T in Pop        // T = (S1,S2)
        Fit(T) = 0;
        f1 = Execute(S1);
        f2 = Execute(S2);
        Fit(T) = f1 + f2;
    New_Pop = { };
    For i = 1 to POP_SIZE/2
        Select P1 and P2 from Pop by Fit;
        Perform Crossover;
        Perform Mutation;
        Insert P1 and P2 into New_Pop;
    Pop = New_Pop;
Tbest = Best(Pop);
```

Since we desire smaller program [9], a complexity term was used in all experiments to penalize large trees:

$$F = F + \beta C \quad (3)$$

Symbol	Description
X_r	x-axis distance between target and robot r
Y_r	y–axis distance between target and robot r
S_r	number of steps moved by robot r
C_r	number of collisions made by robot r
M_r	distance betw. starting and final pos. of r
A_r	penalty for moving away from other robots
c_i	coefficient for factor i (e.g., $c_1 = 5$)
K	positive constant (e.g., $K = 40$)

Table 1: Symbols used for fitness definition.

where C is the number of nodes in the tree and β is a smll constant (e.g., 0.0001).

4 Genetic Programming with Data Selection

The fitness, $F_i(g)$, of program i at generation g is measured as the average of its fitness values $f_{ij}(g)$ for the cases j in the training set.

$$F_i(g) = \frac{1}{S}\sum_{j=1}^{S} f_{ij}(g) \tag{4}$$

where S is the number of fitness cases.

For the initial population, a small subset of fitness cases, D_0 is chosen from the base training set $D^{(N)}$ of size N

$$D_0 \subset D^{(N)}, \quad |D_0| = n_0. \tag{5}$$

After individuals are evolved by the usual evolutionary process (fitness evaluation, selection, and mating to generate offsprings), a portion of training set, d_g, is chosen randomly from the previous candidate set C_{g-1}

$$d_g \subset C_{g-1}, \quad |d_g| = \lambda, \tag{6}$$

where $C_{g-1} = D^{(N)} - D_{g-1}$. And it is mixed with the previous training set to make a new training set D_g for the next generation

$$D_g = D_{g-1} \cup d_g, \quad D_{g-1} \cap d_g = \{\}. \tag{7}$$

That is, the sequence of training sets for GP active is

$$D_0 \subset D_1 \subset D_2 \subset ... \subset D_G = D^{(N)}, \tag{8}$$

where G is the number of max generation.

Since GP with active data selection uses only a subset of given data accumulated at each generation to evolve programs, while GP standard use all the given training data repeatedly, the number of fitness cases at each generation is

$$S_g = \begin{cases} N, & \text{for GP standard} \\ n_0 + \lambda g & \text{for GP active.} \end{cases} \tag{9}$$

Parameter	Value
Terminal set	`FORWARD, AVOID,`
	`RANDOM-MOVE, TURN-TABLE`
	`TURN-GOAL, STOP`
Function set	`IF-OBSTACLE, IF-ROBOT,`
	`IF-TABLE, IF-GOAL`
	`PROG2, PROG3`
Fitness cases	100 training worlds,
	100 test worlds
Robot world	32 by 32 grid, 64 obstacles,
	1 table to transport
Population size	100
Max generation	200
Crossover rate	1.0
Mutation rate	0.1
Max tree depth	10
Selection scheme	truncation selection with elitism

Table 2: Tableau for the table transportation problem.

5 Experiments and Results

Table 2 summarizes the experimental setup for genetic programming. The terminal and function set consists of six primitives, respectively. Each fitness case represents a world of 32 by 32 grid on which there are four robots, 64 obstacles, a table to be transported. A total of 100 training cases are used for evolving the programs for standard GP. The GP run with active data selection use $10 + 3g$ examples out of the given data set, i.e. $n_0 = 10, \lambda = 3$, for fitness evaluation. For both methods, a total of 100 independent worlds were used for evaluating the generalization performance of evolved programs.

Figure 2 shows the result of fitness switching method for evolving composite cooperative behaviors. Figure 3 shows the fitness of two methods with repect to the number of evaluations calculated as a product of the population size and the data size for each generation. The GP active achieved a speed-up factor of approximately two compared with that of the standart GP. The results are summarized in Table 3. Though the GP with active data selection used a smaller set of fitness cases, its training and test performance were slightly better than those of the standard GP.

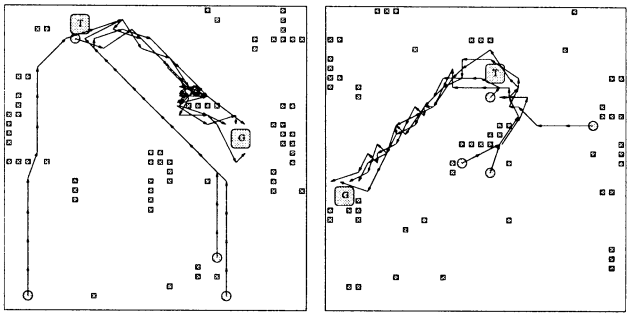


Figure 2: Trajectory of robots running the evolved program in the training and test cases.

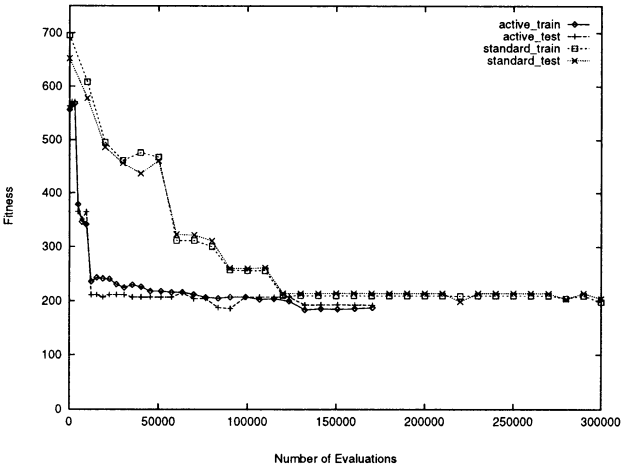


Figure 3: Comparison of fitness values as a function of the number of function evaluations.

Method	Time	Average Number of Steps	
		Training	Test
GP standard	150000	209.562	213.839
GP active	76500	206.654	203.996

Table 3: Comparison of time and average fitness values for the standard GP and the GP with active data selection.

6 Conclusions

Using the table transport problem we have experimentally shown that the fitness switching method can evolve composite cooperative behaviors of multiple robotic agents. Experimental results also show that by reducing the effective number of fitness cases through active data selection the evolution speed of GP can be enhanced without loss of generality of evolved programs.

Acknowledgments

This research was supported in part by the Korea Science and Engineering Foundation (KOSEF) under grant 96-0102-13-01-3.

References

[1] Werner G. M. and Dyer M. G. 1993. Evolution of herding behavior in artificial animals. In *Proceedings of Second International Conference on Simulation of Adaptive Behavior.* Pages 393-399.

[2] Koza, John R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* The MIT Press.

[3] Bennett III, Forrest H. 1996. Automatic creation of an efficient multi-agent architecture using genetic programming with architecture-altering operations. In J.R. Koza *et al.* (eds.), *Proceedings of the First Annual Conference on Genetic Programming.* The MIT Press. Pages 30-38.

[4] Luke, Sean and Spector, Lee. 1996. Evolving teamwork and coordination with genetic programming. In J.R. Koza *et al.* (eds.), *Proceedings of the First Annual Conference on Genetic Programming.* The MIT Press. Pages 150-156.

[5] Iba, Hitoshi. 1997. Multi-agent learning for a robot navigation task by genetic programming. In J.R. Koza *et al.* (eds.), *Proceedings of the Second Annual Conference on Genetic Programming.* Morgan Kaufmann. Pages 195-200.

[6] Luke, Sean. 1998. Genetic Programming Produced Competitive Soccer Softbot Teams for RoboCuy97. In J.R. Koza *et al.* (eds.), *Proceedings of the Third Annual Conference on Genetic Programming.* Morgan Kaufmann. Pages 214-222.

[7] Zhang, B. T. and Cho, D. Y. 1998. Fitness Switching: Evolving Complex Group Behaviors Using Genetic Programming. In J.R. Koza *et al.* (eds.), *Proceedings of the Third Annual Conference on Genetic Programming.* Morgan Kaufmann. Pages 431-438.

[8] Zhang, B. T. and Veenker, G. 1991. Focused incremental learning for improved generalization with reduced training sets, *Proc. Int. Conf. Artificial Neural Networks*, Kohonen, T. *et al.* (eds.) North-Holland, Pages 227-232.

[9] Zhang, B. T., Ohm, P., and Mühlenbein, H. 1997. Evolutionary induction of sparse neural trees. *Evolutionary Computation.* 5(2) 213-236.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Digital Artificial Brain Architecture for Mobile Autonomous Robots

Andrés Pérez-Uribe* and Eduardo Sanchez
Logic Systems Laboratory, Computer Science Department
Swiss Federal Institute of Technology-Lausanne
CH–1015 Lausanne, Switzerland
{Andres.Perez,Eduardo.Sanchez}@di.epfl.ch, http://lslwww.epfl.ch/

Abstract

An autonomous robot need not be given all the details of the environment in which it is going to act: it can acquire them by direct interaction. One approach to learn by interaction is *reinforcement learning*, though, the robot has also to be able to autonomously categorize the input data it receives from the environment, deal with the stability-plasticity dilemma, and learn very rapidly. In this paper we present a *digital artificial brain architecture* capable of dealing with such problems. Furthermore, we present its use for controlling a mobile autonomous robot in an obstacle avoidance task in a real arena.

Keywords. Artificial neural networks, mobile autonomous robots, neurocontrol.

1 Introduction

Programming an autonomous robot so that it reliably acts in an unknown or a dynamic environment is a difficult thing to do. This is due to missing information during programming, the dynamic nature of the environment and the inherent noise in the robot's sensors and actuators [1].

One common approach to the control of autonomous robots is that of *reactive systems*. However, such approaches may not generate good solutions to the navigation problem since the control system reacts as a function of the sensor's readings and the robot's perception is limited. An interesting approach it to add learning capabilities to reactive systems [2].

An autonomous robot need not be given all the details of the environment in which it is going to act: it will acquire them by direct interaction and extrapolation [1]. To achieve this, the robot must be controlled by an artificial brain, able to autonomously categorize the input data it receives from its environment. It has to deal with the *stability-plasticity* trade-off (i.e., how can it preserve what it has previously learned, while continuing to incorporate new knowledge). And finally, it has to be able to generalize between similar situations, and develop a proper policy for action selection, based on an evaluative reinforcement signal received from the environment.

In this paper we present the simulation and test of a digital architecture of an artificial brain for obstacle avoidance in mobile autonomous robots (see figure 2). Our main objective is to implement it using programmable hardware devices (i.e., Field-Programmable Gate Array [3] devices). Therefore, herein we describe the resulting architecture, and present some results of the simulation of the system considering the hardware constraints. Basically, we considered 8-bit precision and accounted related problems like overflow, underflow and the limited precision in arithmetic computations.

2 Background and experimental setup

The development of learning techniques for mobile autonomous robots constitutes a major field of research. A lot of work has been done running simulated robots in simulated environments, though sometimes very nice results obtained in such way do not hold in the real world [1]. More recent research plans to use simulation to speed up initial adaptation of the controller [4]. *Evolutionary techniques* have been successfully used to adapt a robot controller while considerably reducing human design, though, the duration of experiments is rather long [4]. Our approach is intended to speed up adaptation by means of learning by interaction instead of evolution, and is closely related to the works of Bruske et al. [5] and C. Scheier [6] based on *dynamic cell structures* of RBF units, and Millan's incremental learning approach [2] based on

* *Pérez-Uribe is supported by the Centre Suisse d'électronique et de Microtechnique CSEM, Neuchâtel, Switzerland.*

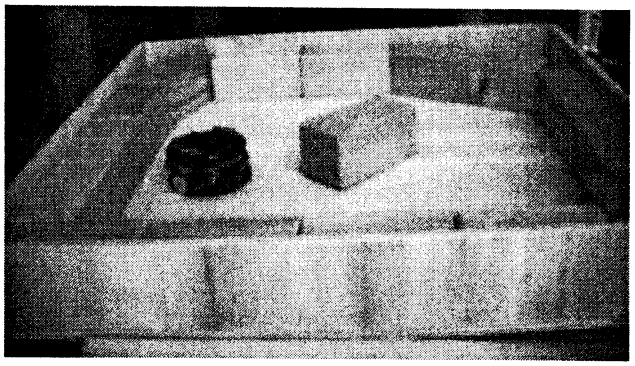


Figure 1: Obstacle avoidance task.

Alpaydin's GAL supervised learning model [7].

We have used the Khepera mobile autonomous robot [8] connected to a host computer by a RS-232 serial port, to test our digital brain architecture in an obstacle avoidance task in a real arena made of wood (see figure 1). The Khepera is provided with eight 10-bit precision infra-red sensors, six (IR0 to IR5) on one side of the robot (front) and the remaining two (IR6 and IR7) on the other side (back). We used as inputs to the neurocontroller, three 8-bit values: L (left), R (right) and B (back), corresponding to a preprocessing of the eight infra-red sensor signals: L = max(IR0,IR1,IR2), R = max(IR3,IR4,IR5) and B = max(IR6,IR7). The possible set of actions were *go forward*, *go left* and *go right* with a fixed speed of 40mm/s.

3 Artificial brain architecture

To address the problem of adaptive categorization, we used the Flexible Adaptable-Size Topology neural network (FAST) [9], an unsupervised learning neural network that *clusters*, *codes*, or *categorizes* input data by activating the same *neuron* when the inputs are *sufficiently* similar, and dynamically determines the number of categories in the input data by changing its size (i.e., the number of neurons) by a *growing* (when an input vector is *sufficiently* different) and a probabilistic *pruning* mechanism.

The FAST architecture is a direct adaptation of the Grow and Represent (GAR) algorithm [7], which in turn is an extension of the *ART* algorithm [10], motivated by the possibility of a hardware implementation [9]. In our approach, FAST dynamically categorizes the 3-dimensional infra-red sensor signals (L,R,B), serving as a sparse-coarse-coded function approximator of the input state space [11], providing generalization and at the same time, dealing with the stability-plasticity dilemma (by locally modifying the weights). The system has been design to support 8-bit computation (see [9] for more details on the hardware implementation), and a maximum number of 40 categories was selected for this application.

To learn a *reactive* system by interacting with the environment based on an external evaluative signal, we used reinforcement learning techniques [12]. Reinforcement learning selectively retains the actions that maximize the received reward over time. In particular, we implemented an adaptation of the Dyna-SARSA algorithm (an integrated architecture for Learning, Planning and Reacting) [13].

Basically, SARSA [14] attempts to solve the *temporal-credit assignment dilemma*, i.e., how to punish or reward an action when it might have far reaching effects, by updating reinforcement estimates (Q values) using previously learned estimates (i.e., by *bootstrap*) as follows:

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma Q(s',a') - Q(s,a)],$$

where s is the current state of the environment (determined in our approach by the sparse-coarse-coded function approximator), a is one of the three possible actions, and s' and a' are the corresponding possible next state and action. r is the evaluative reinforcement signal (-1 if any of the infra-red sensors exceeds a certain threshold, 0 otherwise). In the case of Dyna-SARSA, a Q action value update is performed after every interaction with the environment, plus a certain number of times using a dynamically generated probabilistic model of the environment, which makes the system to learn faster and more efficiently [15].

Finally, the presented neurocontroller architecture includes a pre-wired *basic reflex* codified as a simple reactive behavior: follow the direction of the less activated sensors during 10 times 10 time steps. This basic reflex is activated when any of the infra-red sensor exceeds a certain threshold value.

We are currently working on the digital implementation of the Dyna-SARSA algorithm, based on a previous implementation of the Adaptive Heuristic Critic (AHC) algorithm [16]. The new implementation has also been conceived for 8-bit precision computation, but this time, we use a single action-value (Q) instead of separated values for the *actor* and the *critic*, and model-based updates of the action-values to accelerate learning instead of using eligibility traces [17], which are computationally intensive [18].

The target programmable device for the whole neurocontroller system is a Xilinx XC40125 [19], with ap-

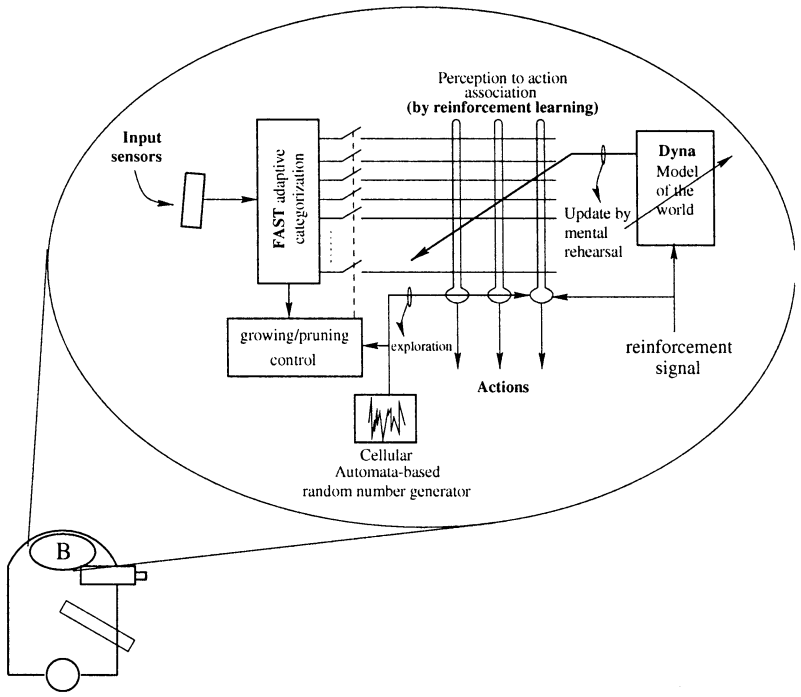


Figure 2: Artificial brain architecture.

proximately 125,000 equivalent gates, that is going to be integrated to the Khepera robot.

4 Discussion and results

We have successfully simulated the complete artificial brain architecture considering digital hardware constraints. Basically, we considered 8-bit precision and accounted related problems like overflow, underflow and the limited precision in arithmetic computations. The described artificial brain was used to control the Khepera mobile autonomous robot in a real arena (see figure 1). The robot successfully learned an appropriate state-space partition (adaptive categorization) and a perception-to-action mapping such that the mobile robot was able to avoid obstacles by staying in a quasi-circular path around the obstacle in the center of the arena. Furthermore, the neurocontroller was able to generalize its mappings when we slightly changed the position or the form of the obstacle in the center of the arena.

While FAST is not the first neural network based on adaptable topology, it is unique in that it does not require intensive computation to reconfigure its structure, and can thus exist as an on-line learning stand-alone machine which could be used, for example, in real-time control applications, like dynamic categorization in autonomous robotics.

Dyna has proven to be a very useful mechanism to enhance reinforcement learning techniques like SARSA, and to be as useful as learning with eligibility traces in Maze learning tasks (a commonly used abstraction of the real navigation problem) [18]. We implemented Dyna because it has the appealing property of accelerating learning by a *mental rehearsal* mechanism (i.e., the repeated presentation of memorized previous experiences), and because from the computational point of view, it is less intensive than eligibility traces, and can be executed at electronic speeds between consecutive interactions of the robot with the environment.

One further point was the partial observability of real environments: in mobile robotics, the agent has only partial information about the current state of the environment, that is, it does not know the state of the whole world from the state of the sensory input alone (i.e., the observations). The agent is said to suffer from *hidden-state* [20] or *perceptual aliasing* [21]. Dyna appeared to be quite sensitive to partial observability in Maze learning tasks [18]. However, this issue was partly solved by function approximation and incremental construction of the state representation by means of the dynamic categorization. We plan to in-

troduce a new fast learning representation while simultaneously expanding it over time by means of short-term memories to disambiguate identical sensory inputs observed in different states.

5 Conclusions

We have presented an integrated neurocontroller architecture (i.e., an artificial brain architecture) for mobile autonomous robots. The given architecture has been motivated by the possibility of digital implementation using programmable devices, and provides a learning robot the capability to autonomously categorize input data from the environment, deal with the stability-plasticity dilemma and learn a perception-to-action mapping that enables it to navigate straight forward while avoiding obstacles.

References

[1] M. Dorigo. Editorial introduction to the special issue on learning autonomous robots. In *IEEE Trans. on SMC-Part B*, pages 361–364, June 1996.

[2] J. del R. Millan. Rapid, safe, and incremental learning of navigation strategies. In *IEEE Trans. on SMC-Part B*, pages 408–420, June 1996.

[3] S.M. Trimberger, *Field-Programmable Gate Array Technology*, Kluwer Academic Publishers, Boston, 1994.

[4] D. Floreano. Reducing Human Design and Increasing Adaptability in Evolutionary Robotics. In *Evolutionary Robotics*, T. Gomi (Ed.), AAI Books, Ontario, 1997.

[5] J. Bruske and I. Ahrns and G. Sommer. An integrated architecture for learning of reactive behaviors based on dynamic cell structures. *Robotic and Autonomous Systems*, 22, pages 87-101, 1997.

[6] C. Scheier. Incremental Category Learning in a Real World Artifact Using Growing Dynamic Cell Structures. In *Proceedings of the 4th European Symposium on Artificial Neural Networks ESANN'96*, M. Verleysen (Ed.), pages 117-122, Bruges, 1996.

[7] A. Alpaydin. *Neural Models of Incremental Supervised and Unsupervised Learning.* PhD thesis, Swiss Federal Institute of Technology, Lausanne, 1990. These 863.

[8] F. Mondada and E. Franzi and P. Ienne. Mobile robot miniaturization: A tool for investigating in control algorithms. In *Proceedings of the Third International Symposium on Experimental Robotics*, Kyoto, Japan, 1993.

[9] A. Perez-Uribe and E. Sanchez. FPGA implementation of an adaptable-size neural network. In *Proceedings of the Intl. Conf. on Artificial Neural Networks ICANN96*, pages 383–388, Springer Verlag, July 1996.

[10] G. Carpenter and S. Grossberg. The ART of Adaptive Pattern Recognition by a self-organizing neural network. *IEEE Computer*, pages 77–88, March 1988.

[11] R.S. Sutton. Generalization in reinforcement learning: Successful examples using sparse coarse coding. In *Advances in Neural Information Processing Systems 8*, pages 1038–1044, MIT Press, 1996.

[12] R.S. Sutton and A. Barto, *Reinforcement Learning: An Introduction.*, MIT Press, 1998.

[13] R.S. Sutton. Integrated architectures for Learning, Planning, and Reacting based on approximating Dynamic Programming. In *Proceedings of the Seventh International Conference on Machine Learning*, pages 216–224, Morgan Kaufmann, 1990.

[14] G. Rummery and M. Niranjan, On-line q-learning using connectionist systems. Tech. Rep. Technical Report CUED/F-INFENG/TR 166, Cambridge University Engineering Department, 1994.

[15] L. Kuvayev and R.S. Sutton. Approximation in Model-Based Learning. In *Proceedings of the ICML'97 Workshop on Modelling in Reinforcement Learning*, Vaderbilt University, July, 1997.

[16] A. Perez-Uribe and E. Sanchez. FPGA implementation of a Network of Neuronlike Adaptive Elements. In *Proceedings of the Intl. Conf. on Artificial Neural Networks ICANN97*, pages 1247–1252, Springer Verlag, 1997.

[17] R.S. Sutton. Learning to predict by the methods of Temporal Differences. In *Machine Learning* 3, Kluwer Acad. Pus., pages 9–44, 1988.

[18] A. Perez-Uribe and E. Sanchez. A Comparison of Reinforcement Learning with Eligibility Traces and Integrated Learning, Planning and Reacting. Proceedings of the Intl. Conf. on Computational Intelligence for Modeling Control and Automation CIMCA'98, IOS Press, 1998 (to appear).

[19] "XC4000XV Family Field Programmable Gate Arrays" *Xilinx,Inc* May, 1998

[20] L.J. Lin and T.M. Mitchell. Reinforcement learning with hidden states. *From Animals to Animats: Proceedings of the Second Intl. Conf. on Simulation of Adaptive Behavior.*, J-A. Meyer, H.L. Roitblat, and S.W. Wilson (Eds.), 1992.

[21] S.D. Whitehead and D.H. Ballard. Active perception and reinforcement learning. In *Proceedings of the Seventh Intl. Conf. on Machine Learning*, Austin, 1990.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Complexities in Biosystem

Hiroshi Tanaka
Medical Research Institute, Tokyo Medical and Dental University
1-5-45 Yushima, Bunkyo-ku Tokyo Japan
Email: tanaka@tmd.ac.jp

Abstract

The characteristics of the complex systems approach to life are reviewed in its two aspects: the essential structure of life (basic bio-complexity) and its evolution mechanism (evolutionary bio-complication). Our basic concept is hierarchical transition of organizational complexity level of living systems and emphasis on the role of information-mediated self-organization

Introduction

To begin with the organized session entitled by "Complexity in life", I should like to briefly review the characteristics of the complex systems approach to life. When we deal with the complexity of life, in other words, "bio-complexity", in which we incorporate the meaning of how life appears from the complex system approach, we should first take into account the fact that the life evolves. Hence the question might be divided into its two aspects: the essential structure of life (basic bio-complexity) and its evolution mechanism (evolutionary bio-complication).

The question about the essential structure of the life is, from its generative understanding, related to that about the origin of life or minimum realization of life. The second question is about the mechanism of how the complex system like life evolves, which straightforwardly brings us to the critical investigations on the current biological evolution theory. We separately make a brief review on these two aspects of bio-complexity. In reviewing the complexity of life our basic concept is hierarchical transition of organizational complexity level of living systems and role of information-mediated self-organization. I also take special attention to the recent findings in factual sciences such as molecular evolutionary biology, geophysics and paleontology, because our biocomplexity theory must not contradict the recent knowledge of origin and evolution of life.

Origin of life from complex systems approach

Molecular biology begins to precisely depict the cenancestor of the cellular life. Recent advances in the molecular evolutionary biology have provided us quite important knowledge about the origin of life. Among them, most interesting are inferences and discussions about (1) the cenancestor of life which is supposed to exist before the branching between the Bacteria and Archeae/Eukarya urkingdom, and about (2) the minimum gene for cellular life.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

The Synthesis of Gene with Block Automaton

Makoto Kinoshita and **Mitsuo Wada**
Research group of Complex Systems Engineering,
Graduate School of Engineering, Hokkaido University,
Kita 13 Nishi 8, Kita-ku, Sapporo 060, Japan.

Abstract

We regard a self-replicating process in multi-unit system as a life. Our purpose in this study is to describe the scenario of evolution from 'self-replication' to 'self-reproduction'. 'Self-replication' is a process of resulting in an exact duplication of parent units. The contrasting term, 'self-reproduction,' is a process that has structures of data units and program units. In this paper, we propose a "Block Model" that consists of three-dimensional block structures. As the result, we obtain a emergence of partialities of reaction possibility and sequential reactions. These properties become a key phenomenon of self-reproduction.

1. Introduction

Our main purpose is to construct a system which produces various shape like life by the translation of their structure through the interaction. In the filed of Artificial Life, considerable number of studies have been made on this problem [3, 4, 5, 6, 8, 10, 11, 12], however, we should like to explore it without the genetic method. It is for this reason that if the genotype-phenotype mapping were decided by system designer, the products of this system is already decided too. Therefore all products is restricted within the intent of creator. To avoid this problem we consider the system which is non-genetic at initial state, and we expect that a system emerges a non-immediate and mutable rule which consists of elements of systems. Since this rule would be able to produce new structure made of elements, and to change itself, this system can occur eternal translation of shape.

For this purpose, we have remarked a self-reproduction phenomenon. Because this phenomenon has been considered as the origin of life in biology. Thus we hope to construct a system which evolves from a state in which there are not self-reproduction elements to a state in which self-reproduction phenomenon is possible. The self-reproduction phenomenon is divided into two types [12]. These are called 'self-replication' and 'self-reproduction.' 'Self-replication' is a process of resulting in an exact duplication of parent units. The contrasting term, 'self-reproduction,' is a process that has structures of data units and program units. In other words, this process is genetic process. Our purpose is to understand a scenario of the translation from self-replication to self-reproduction.That is to say, it is to resolve a road of evolution from the self-replication by auto-catalytic RNAs to the genetic self-reproduction by using DNAs.

We have studied the the multi-unit system to construct above system. This system is composed of units and interaction rules. The units can connect and disconnect with others, and reconstruct themselves into new form. The reaction between units is dominated by the interaction rules.

Using this system, we discuss how a process translate from random reaction, via direct self-replication, to genetic self-reproduction. To consider this question we focus attention on a idea of partiality of reaction possibility and emerged sequences of reaction. The reaction possibility is defined as the ratio of units which can be reacted with it to all units. A unit which has high reaction possibility is considered as an activated matter, and it may have a property of reaction stimulator. Therefore we could say that the emergence of a partiality of reaction possibility is said to appear the difference of properties of units. It is also important that the sequence of reaction appears from random reaction. It is the cause of various stable reaction paths which has different speed and size. A number of reaction paths may form the hierarchies of reaction. The emergence of properties and hierarchies are very important phenomenon to make the genetic self-reproduction.

We have developed a block automaton to implement this system. The one of most important character of this model is that a unit possesses a three-dimensional structure. It brings about various interesting properties. Each unit can connect with others, and form new structures. The interaction between units is determined by their local structures. Through the interaction, the number of atoms does not vary, therefore the material is conserved.

In Section 2, the definition of block automaton is described. Section 3 shows two important properties of this system. Section 4 presents results of experiments and discussion.

2. Block Automaton

In this section, the model which we called block automaton is introduced. This model is one implementation of multi-unit systems. The element of this model possesses three-dimensional structures. The state of automaton is represented as the set of connected elements.

2.1 Atom, Bond and Block

A fundamental element of the block automaton is a diagonal cut cube (Figure 1). We call it 'atom.' An atom a has two square surfaces and a rectangular one. Two

atoms can combine through their surface each other.

The matter which glues atoms is called 'bond.' A bond is represented as $\beta(a,a)$ since it connects two atoms. There are two kinds of bond, one is for a square surface, other is for a rectangular one. In this study, the bond for square surface is treated as primary bond. The bond for rectangular is used in interaction. There are four ways of combining of two atoms. The bond for square is represented four alphabets U, R, L, D respectively (Figure 2).

The set of connected atoms and bonds is called 'block.' It is represented with a string of its bonds. Figure 3 shows a block at 3-dimensional space and its description. It is noted that the bond for rectangular surface is occasionally included in block.

This definitions of these elements are, of course, inspired by a composition of proteins. An atom corresponds to amino-acid. A bond for a square surface corresponds to a peptide bond, one for a rectangular surface to a hydrogen bond. And a block corresponds to a protein.

2.2 Interaction Rule

The interaction is represented as formal rule. The rule f decides the recombination way of two blocks, and is shown as $f:\mathbf{L}_a \times \mathbf{L}_b \rightarrow \{\mathbf{B}\}$, where $\mathbf{L}$ means a local block. The rule is defined for all combination of $\mathbf{L}$ respectively.

The way of rule application is show in figure 4. First of all, two blocks are selected. Second, two atoms are choosed from blocks respectively. These atoms are used as the center of the interaction. Then, a rule is applied, and atoms are recombined. Finally, new blocks are reconstructed.

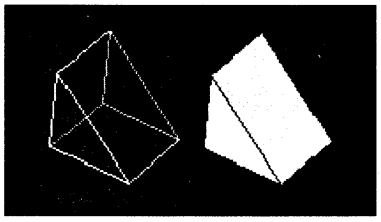


Figure 1. Atom

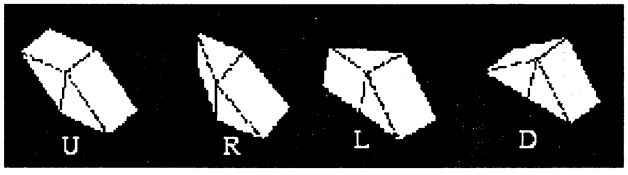


Figure 2. Bonds. Four ways of Combining



Figure 3. Block, and its description.

2.3 Legality of Rule Application

When the rule is applied among two blocks, the legality is checked at 3-dimensional space at two point (figure 4). First of all, it is tested whether two blocks can connect through each diagonal surfaces of targeted atoms. If it is possible to connect, a rule is applied, and atoms are recombined. Finally, it is tested whether they can form new structure without collision.

It must be noted that there are two restriction when rule is applied. These restrictions come from that the units have 3-dimensional structures.

2.4 Neumann Tank

We used an idealized tank as the field of reaction. We call it 'Neumann Tank.' Of course, it is named after John von Neumann who introduced the theory of self-reproduction automaton [8]. A vast number of blocks float in this tank, and interact each other.

The tank $\mathbf{NT}$ is composed of a set of blocks $\{\mathbf{B}\}$, a set of rules $\{\mathbf{F}\}$, and a update schedule of reaction U. Since it is described as $\mathbf{NT}=(\{\mathbf{B}\},\mathbf{F},U)$. We can hold blocks number of size s in a tank. And a tank has two pipes for blocks through in and out.

The reaction process in tank is as follows. First, two blocks are selected under the update schedule U. Second, the interaction occurs among them. After the interaction, the total number of blocks could vary in the tank. Finally, if the number of blocks increases, some blocks flow out, and decreases, some blocks come in. The number of atoms are variable but of blocks are invariable.

3. Properties of Our System

We introduce two properties of our system. We regard these properties have an important role.

3.1 Reaction Possibility

The definition of reaction possibility is described as follows:

$$rp(a)=\frac{\sum_{i=1}^{s}\sum_{j=1}^{len(\mathbf{B}_i)} applicable\left(a,a_j \middle| a_j \in \mathbf{B}_i\right)}{\sum_{i=1}^{s} len(\mathbf{B}_i)}$$

where $len(\mathbf{B})$ means the length of $\mathbf{B}$, and $appliciable\left(a_x,a_y \middle| a_x \in \mathbf{B}_x, a_y \in \mathbf{B}_y\right)$ is 1 if $\mathbf{B}_x$ and $\mathbf{B}_y$ can be applied proper rule through a_x and a_y, or 0 otherwise.

This quantity is calculated for each atom. It represents the possibility of occurrence of reaction between an atom and others. It is necessary to connect two blocks without collision, when they interact. The larger block usually has

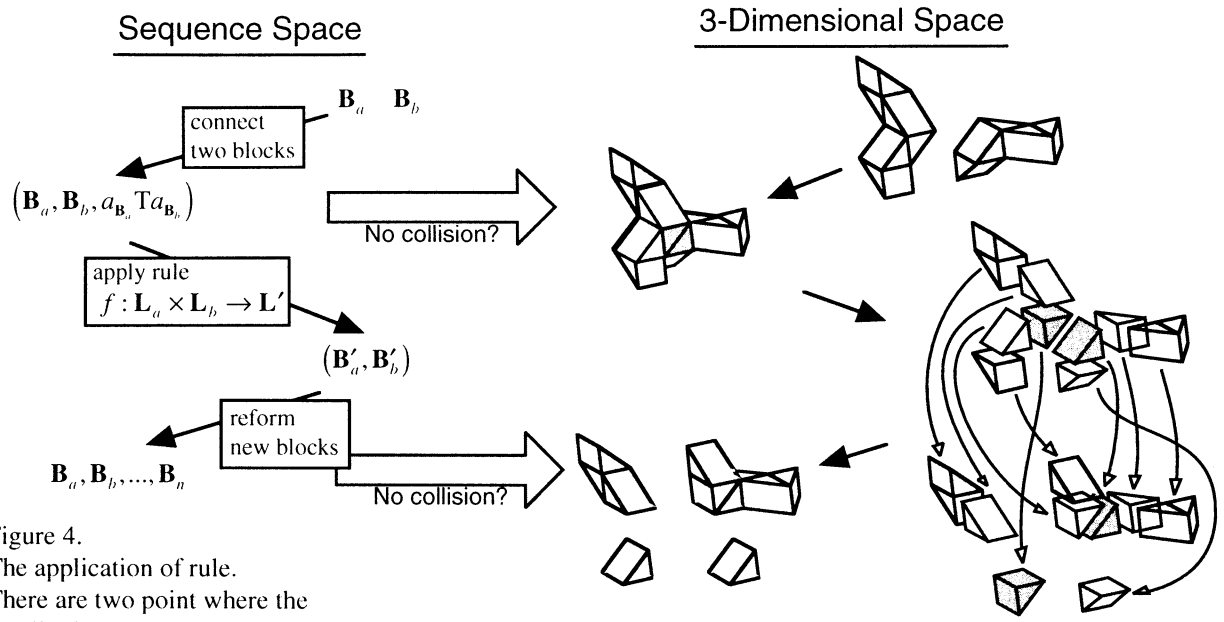


Figure 4.
The application of rule.
There are two point where the legality is tested at 3-dimensional space.

low reaction possibility, because it tends to collide other one.

3.2 Robust Structure of Blocks

In some cases, a part of block of which reaction possibility is 0 appears. We call it preserved local structure. The cause of it is the structure which surrounds itself. By such a structure, the surrounded part of block can not connect other blocks for reaction. Therefore, this structure could remain in the tank.

4. Experiment

In our experiment, we choosed a interaction rule randomly, and simulate reaction in Neumann Tank. The size of Neumann Tank was 100, the range of reaction region was 1. For each time step, we choosed two blocks randomly. Then we applied reaction rule these two blocks. Here atoms for reaction were also choosed randomly in each block.

We analyzed the tank after 10,000 steps, the number of atoms became about 1500 ~ 2000.

4.1 Non-uniform Distribution of Reaction Possibility

First of all, we analyzed a reaction possibility of typical blocks. One example is shown in figure 5. This figure shows one block and reaction possibility of atoms in it. The gray shaded atoms shows that which has the reaction possibility larger than 0. The number of atoms is 18. In this case, only 6 atoms have non-zero reaction possibility. And the highest reaction possibility is 10 times larger than the lowest one without 0 reaction possibility. Therefore, it is said that the number of atoms which can occur reaction are a little in proper length blocks. It may be regarded as the emergence of an active site.

Then a quantity of reaction expectation is defined to show the whole tank. A reaction expectation is average value of reaction possibility of all atoms in a block. A graph of distribution of reaction expectation is shown in figure 6. In it, the block of which length is shorter than 5 is drawed by gray region. This graph indicates that all of blocks which have high reaction expectation are small blocks. It is recognized that a reaction of large blocks are caused by short blocks. A short block could have an ability of reaction stimulator. Therefore, it is said each

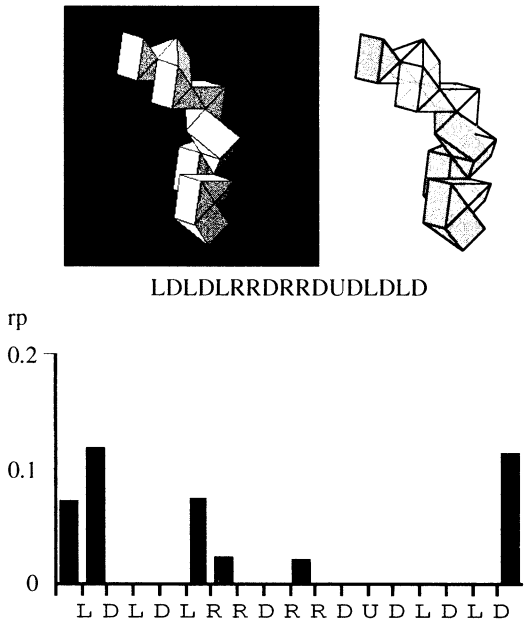


Figure 5. Reaction probabilities of a block.
This figure shows a block and its reaction probabilities under the proper rule and **NT**. The graph shows that the number atoms which have rp larger then 0 are only 6.

block has the different properties depending on its form.

4.2 Sequence of Reaction

The preserved structure could react if the surrounded block has removed. However, we can consider a nested structure like that a preserved part of the block is surrounded by other preserved part of the block is surrounded... and so on. To decompose this nested structure, the sequence of reaction is required.
Example of sequential reaction is shown in figure 7. In this figure, the enclosed region with gray line indicates a preserved structure. To decompose this block, the sequence of reaction, (a) to (b) and (b) to (c), is required. This sequence of reaction is one of the emergence of the context from random reaction.

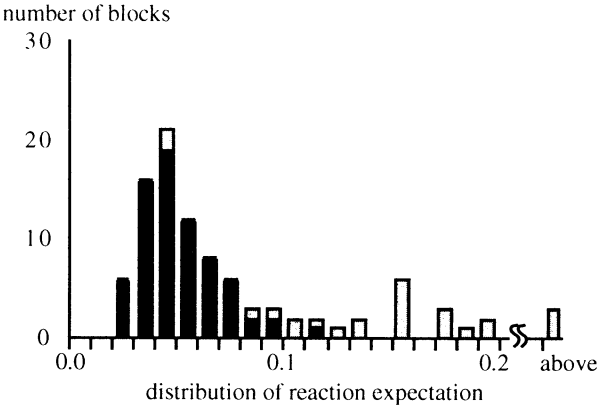


Figure 6. Distribution of reaction expectations of NT. The gray region expresses the blocks shorter than 5.

Summary

We have developed the block automaton for studying the origin of self-reproduction using gene. We defined the value of a reaction possibility and the structure of a preservation to analyze reactions. As the result of several experiments, we observed the emergence of a property depending on the partiality of reaction possibility, and a reaction sequence depending on the preservation.

Acknowledgements

The author benefited greatly from numerous conversations with A. Yamaguchi, M. Kubo, and S. Mikami.

References

1. B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, J.D. Watson, *Molecular Biology of The Cell*, 1994.
2. N. A. Baas, "Emergence, hierarchies and hyper-structures," *Artificial Life III*, 1993.
3. R.J. Bagley, J.D. Farmer, "Spontaneous emergence of a metabolism," *Artificial Life II*, 1992, 93-140.
4. J. Byl, "Self-reproduction in small cellular automata," *Physica D*, Vol 34, 1989, 295-299.
5. W. Fontana, "Algorithmic chemistory," *Artificial Life II*, 1992, 159-209.
6. C. G. Langoton, "Self-reproduction in cellular automata," *Physica D* ,Vol 10, 1984, 135-144.
7. B. Mayer, S. Rasmussen, "Self-Reproduction of Dynamicalk Hierarchies in Chemical Systems," *Artificial Liefr VI*, 1998.
8. J. von Neumann, *Theory of Self-Reproducing Automata*, University of Illinois Press, 1966.
9. J. Y. Perrier, M. Sipper, J. Zahnd, "Toward a viable, self-reproducing universal computer," *Physica D*, Vol 97, 1997, 335-352.
10. S. Rasmussen, C. Knudsen, R. Feldberg, "Dynamics of programmable matter," *Artificial Life II*, 1992, 211-251.
11. T. S. Ray, "An approach to the synthesis of life," *Artificial Life II*, 1992, 371-408.
12. M. Sipper, et al. The POE Model of Bio-Inspired Hardware Systems: A Short Introduction, *Genetic Programming 1997*, 1997.
13. H. Sayama, "Introduction of Structual Dissolution into Langton's Self-Reproducing Loop," *Artificial Life VI*, 1998.

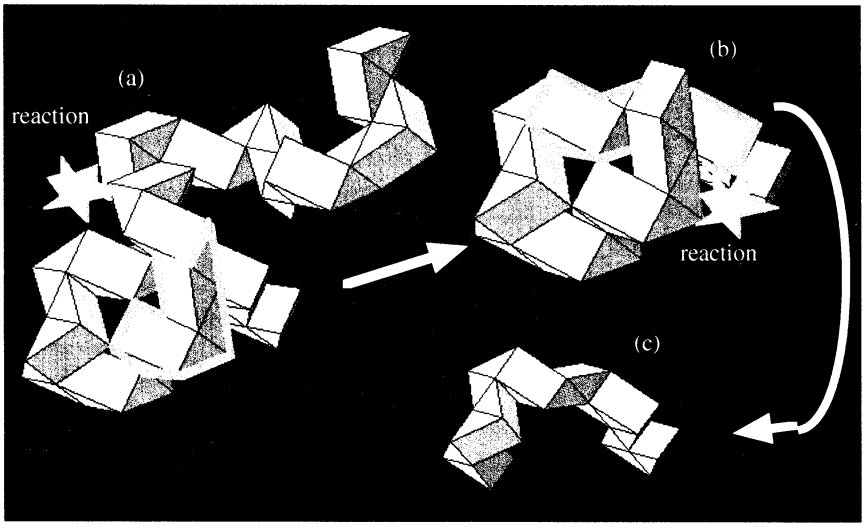


Figure 7. One example of sequential reaction path. The region enclosed with gray line means the preserved structures.
(a) First, there was a long block which had some preserved part of the block. A reaction occurred, and the part which was cause of the preservation removed.
(b) The part which was preserved at (a) became into the possible one for reaction. And next reaction occurred.
(c) At last, the part in the deepest came to out.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Statistical Approach to Genetic Algorithm

Yoshi Fujiwara*
Auditory and Visual Informatics Section, Communications Research Laboratory,
Iwaoka 588-2, Nishi, Kobe 651-2401, Japan

Abstract

In this manuscript, a statistical approach of genetic algorithm, based on macroscopic variables of fitness distributions and averaging out microscopic variables of genotype configurations, is reviewed following the work by Prügel-Bennett, Shapiro and others.. This is a useful viewpoint when one plans and modifies GA parameters and algorithms, not generically but in a problem-specific way. In a simple example for illustration, we show that a general form of fitness which is written in terms of additive bit-strings can be treated in an series-expansion form in the case of mutation-selection dynamics. In the talk, we present a preliminary step towards understanding how natural evolution might be viewed as doing computation in the sense of incorporating environmental "information" into the microscopic states of genetic pool of population.

1 Introduction

Genetic Algorithm (GA) has been applied to many practical problems as stochastic search and optimization techniques [1][2]. As well, it provides an alternative tool to model natural evolution including problems of learning effect, sexual selection, ecosystems, to name a few (see e.g. [3]). The algorithm is unique in the population-based parallel algorithm and the genetic operators. Theoretical understanding and its utilization, however, is still in infancy due to the unique stochastic process of GA.

In engineering, planning parameters such as mutation/crossover rates and population size, and choosing schemes such as selection and crossover are of practical importance. In scientific modeling of evolution, it is of interest to understand metastability, finite-size effect and resulting fluctuations, for example. If one argues that evolution is not a mere optimization process, one would need to study dynamical aspects of genotype/phenotype/fitness spaces themselves.

In either cases, it is crucial to know what can be generally argued and what depends on problems one is concerned with. Although some general aspects are clarified by Markov chain analysis (see [4] for example), there are too few powerful tools of theoretical analysis. Rather than a problem-independent "general theory", one would need theoretical tools to separate problem-specific aspects of GA from general ones.

Prügel-Bennett, Shapiro and Rattray have extensively developed a theoretical formalism of GA [5]–[7]. It is motivated by statistical mechanics in physics but, basically, is a rather standard analysis in statistics. The formalism predicts statistical quantities such as mean and variance of fitness distribution in the population and their time evolution.

An important step was to focus on phenotype space, or usually fitness space, distribution by averaging out most "microscopic" degrees of freedom in genotype space. A population with a finite size is then regarded as a random sampling from the distribution. The distribution and its stochastic time-evolution is reduced to its cumulants, giving a finite-dimensional "macroscopic" variables. It is noteworthy that one can analyze selection effect quite generally because selection depends only on the fitness distribution in a population. Recently, van Nimwegen, Crutchfield and Mitchell [8], sharing a similar idea but in a different approach, study metastable stasis in population fitness distributions in the so-called royal road GA modeling the neutral theory of evolution. This gives insights into the nature of time-scales of stasis and innovations between them.

In this manuscript, the essential idea is revisited hopefully in an accessible way to people in the engineering field. The emphasis is on the macroscopic/microscopic viewpoints. We show a simple example and that a general form of fitness which is written in terms of additive bit-strings can be treated in higher-order expansion of selection dynamics. Mutation-selection dynamics with quadratic potential problem will be used for illustration and we see a quite satisfactory agreement between simulations and theory. In the talk, we shall further focus on information-theoretic approach including max-

*Domestic Research Fellow. e-mail: yfujiwar@crl.go.jp

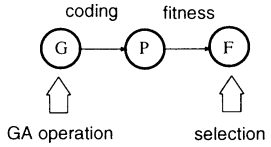


Figure 1: Genotype(G), phenotype(P) and fitness(F) spaces.

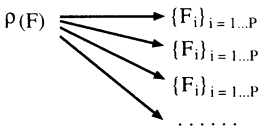


Figure 2: Distribution and different random sampling (realizations) of a population.

imum entropy assumptions, entropy and correlation in microscopic variables of genotype space, based on the ideas given in the manuscript. This is a preliminary step towards understanding how natural evolution might be viewed as doing computation in the sense of incorporating environmental "information" into a genetic pool of population.

In section 2, we describe the basic idea of macroscopic/microscopic viewpoints. In section 3, we give a simple example of a potential problem of additive bitstring. In section 4, we conclude and briefly discuss further extension.

2 Macroscopic Variables

The basic idea is quite natural in the viewpoint of statistics. One performs simulations with different initial populations and obtains many different realizations in the stochastic process of GA. Those different realizations give an *ensemble of finite populations*. One needs some statistical quantities in order to compare the theory with the simulations performed.

Let us recall that the standard viewpoint of genotype, phenotype and fitness spaces in GA (see Figure 2). A natural statistics in GA would be the *fitness distribution* (or phenotype distribution, in some cases) of population.

This leads us quite naturally to model the ensemble of finite populations with a *distribution*. Each realization can then be regarded as a set of P individuals independently sampled from a distribution $\rho(F)$ (see Figure 2). Here and hereafter we denote the size of population by P and the fitness of each individual by F_i $(i = 1, \cdots, P)$.

This viewpoint is familiar as the "population and sample" in statistical inference. It gives us an important bridge between a theory and an experiment. For example, the sample variance of the fitness distribution in a population can be given by $\kappa_2 = (1/P)\sum_i^P F_i{}^2 - (1/P^2)(\sum_i^P F_i)^2$. One takes the average $\overline{\kappa}_2$ over many realizations of populations. On the other hand, it is well known in statistics that the averaged variance over all the different samplings of each set $\{F_i\}_{i=1,\cdots,P}$ from a distribution is related to the variance of the distribution as

$$\overline{\kappa}_2 = \left(1 - \frac{1}{P}\right) K_2, \qquad (1)$$

where K_2 is the variance of the distribution $\rho(F)$.

If one can assume that the fitness distribution does not deviate much from a Gaussian distribution*, then one can characterize such a distribution by using moments and cumulants statistics. In this case, cumulants give a good approximation of dynamics in terms of these macroscopic variables because all the higher order cumulants K_n $(n \geq 3)$ vanish for a Gaussian distribution. K_1 is the mean and K_2 is the variance. K_3 and K_4 represent skewness and kurtosis of the distribution respectively. Note the notation to distinguish the cumulants K_n of distribution from the sample cumulants κ_n in a realization of a population and its ensemble average $\overline{\kappa}_n$.

The above picture has practical advantage over the starting finite-population picture. Generally speaking, distributions are much simpler to handle than finite populations. This is due to the fact that as one tries to calculate the statistical quantities one necessarily has to treat a strong constraint appearing in the corresponding probability distribution. Since the idea of ensemble and distribution allows us to sample a set of members *independently*, the calculation can be extremely simplified and transparent†.

Indeed, the sample moments $\overline{F_i{}^n} = (1/P)\sum_{i=1}^P F_i{}^n$ can be given by the generating function

$$\phi(\xi) = \left\langle\, e^{\xi F} \,\right\rangle_F = \frac{1}{P}\sum_{i=1}^{P} \left\langle\, e^{\xi F_i} \,\right\rangle_{F_i}, \qquad (2)$$

*This assumption is valid in many practical situations in GA with some important exceptions including the innovation process such as [8].

†This is apparently analogous to the relation between the canonical ensemble and micro-canonical ensemble in statistical mechanics, the latter of which has to handle with the constraint for the total energy of a system.

where the last equality comes from the fact of random sampling. Taking logarithm of (2) as usual, one obtains the generating function for cumulants:

$$G(\xi) = \log \sum_{i=1}^{P} \langle\, e^{\xi F_i} \,\rangle_{F_i} + \text{const.} \qquad (3)$$

One can now easily see that this gives a powerful tool for analyzing selection effect on distribution. Selection can be performed in various ways. Simple stochastic selection would be to select each individual (for successive genetic operations) with a probability proportional to its fitness:

$$p_i = w(F_i) / \sum_{i}^{P} w(F_i) \qquad (4)$$

where $w(F_i)$ is a weight for choosing an individual with fitness F_i. (See Introduction for other selection schemes that can be similarly analyzed). Then the cumulant generating function (3) can be written as

$$G(\xi) = \log \sum_{i=1}^{P} w(F_i) e^{\xi F_i} + \text{const.} \qquad (5)$$

Such a generating function is familiar in physics as free-energy, and can be analytically and numerically calculated using techniques in statistical mechanics [7].

On the other hand, since genetic operations act on genotype space (Figure 1), its effect on fitness distribution should be examined in a problem-dependent way. When phenotype is additive in terms of bit-strings without epistasis (in quantitative genetics, the case of additive genotypic value with a negligible environmental deviation to give a phenotypic value), some problems can be explicitly treated. Even in other general problems, how microscopic states in genotype space are distributed under the condition that the macroscopic variables are given may be deduced from maximum entropy assumption. This will be discussed in the last section. Generalization to multi-dimension of fitness space can be also formalized [9].

3 Example for Illustration

Consider a problem in which phenotype is given by additive values of alleles. We assume a bit string for each gene, whose allele s_α can take either 0 or 1. We denote the αth allele of the ith gene by $s_{i\alpha}$. Letting the length of the bit string be L, the phenotype value M_i of the bit-counting problem is given by

$$M_i = \sum_{\alpha}^{L} s_{i\alpha=1}. \qquad (6)$$

The fitness is given as a "potential" function: $F_i = V(M_i)$.

We consider mutation-selection dynamics. Selection is done as described in the preceding section with the weight (4) given by

$$w(F(M)) = \exp(-\beta V(M)), \qquad (7)$$

where β is a selection pressure parameter.

Let us examine the distribution in phenotype space since it is easy to convert it to fitness space distribution by $F = V(M)$. The asymptotic form in the weak selection and large population [7] of the generating function (5) can be given by

$$G(\xi) = \frac{K_1}{\sqrt{K_2}}\bar{\xi} + \log \varphi_1(\bar{\xi}) - \frac{1}{2P}\frac{\varphi_2(\bar{\xi})}{\varphi_1(\bar{\xi})^2}. \qquad (8)$$

Here we used the rescaled variables, $\bar{\xi} = \sqrt{K_2}\xi$, $\mu = (M - K_1)/\sqrt{K_2}$ and

$$\varphi_n(\bar{\xi}) = \int d\mu\, \overline{p}(\mu) \exp(n\xi\mu - n\beta V(\mu)), \qquad (9)$$

where $\overline{p}(\mu)d\mu = \rho(M)dM$. From the generating function (8), one can obtain the selection effect as

$$K_n{}^{\mathrm{s}}/K_2^{n/2} = \frac{\partial^n}{\partial \bar{\xi}^n} G(\xi) \qquad (10)$$

where $K_n{}^{\mathrm{s}}$ is the cumulant after selection. It is straightforward to calculate (10) for general form of $V(M)$ by series expansion in M so that one can obtain higher-order calculation for the selection dynamics [9].

As an example, consider a quadratic potential, $V(M) = (M - M_0)^2$, which was studied in the context of subset-sum problem in [6]. The above calculation of selection dynamics gives a higher-order extension of it. Point-mutation is assumed here: each allele $s_{i\alpha}$ is mutated or "flipped" with a probability γ. Mutation dynamics can be easily obtained in this case as given in [6][7], and even to higher-order [10] [9]. Combining the mutation dynamics with the selection above, we can follow the time-evolution.

Figure 3 is the simulation result for $P = 100$, $L = 64$, $M_0 = L/2$, $\beta = 1/L^2$, $\gamma = 1/L$. The agreement is quite satisfactory in the third and even in the fourth cumulants thanks to the higher-order expansion[‡].

4 Conclusion with Loose End

The essential idea in macroscopic/microscopic viewpoints is revisited following [5]–[7]. We show that

[‡] We explicitly mention that in [6] third-order calculation is given for the quadratic potential problem.

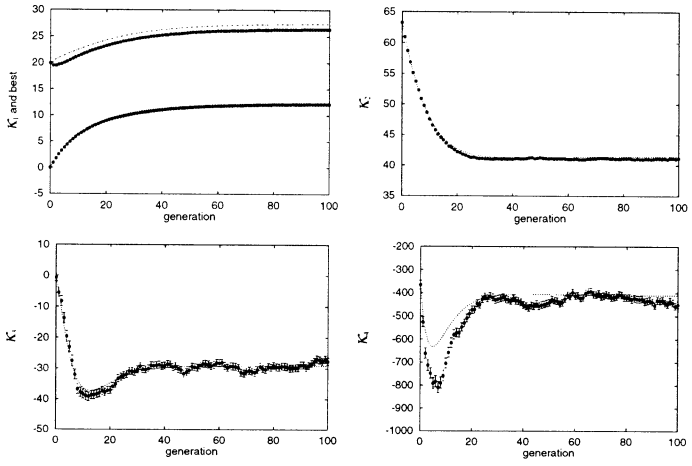


Figure 3: The best M_i in the population, $\overline{\kappa}_n$ ($n = 1, 2, 3, 4$) are plotted versus generation. Error bars represent standard error estimation. The dotted line is the theoretical prediction. See text for parameters.

in the case of general fitness which is a function of additive bit-strings, one can, at least in principle, calculate arbitrary higher-order dynamics of selection in the asymptotic limit of weak selection and large population size. As an example, quadratic potential problem that is related to a NP-hard problem was treated up to fourth-order cumulants and compared with simulations. Due to the limit of space, we did not go into the details but described only on the macroscopic variables in fitness/phenotype space distribution, in the present manuscript. In the talk, we further focus on information-theoretic approach including maximum entropy assumptions, entropy and correlation in microscopic variables of genotype space, based on the ideas given in the manuscript.

To be specific, diversity maintenance in GA is of our interest. In order to discuss it, one has to go into the details of coding (Figure 1), microscopic variables such as how the allele values are distributed in the gene pool of the population, and how they are related to coding. To give an example in this direction, recently proposed thermodynamical rule in GA (see [11] and references therein) is interesting. A good measure for diversity should reflect how allele sites are correlated to each other. It seems that maximal entropy method [7][6] may provide a useful tool to discuss it. We think one can go further into information-theoretic, computational and emergence viewpoints if one regards natural evolution not a mere optimization but as a "computation" to incorporate environmental "information" into the microscopic states of genetic pool of population [12][13].

Acknowledgements

I would like to thank A. Prügel-Bennett and J. Shapiro for for their help and patience. I am grateful to H. Sawai for encouragement. I have benefited from a short conversation with K. Kaneko on [8]. This work is supported by Domestic Research Fellowship of Japan Science and Technology Corporation.

References

[1] J. H. Holland, *Adaptation in Natural and Artificial Systems*, (The University of Michigan Press, 1975; MIT Press, 1992).

[2] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning* (Addison-Wesley, 1989).

[3] M. Mitchell, *An Introduction to Genetic Algorithms*, (MIT Press, 1996).

[4] A. E. Nix and M. D. Vose, "Modeling genetic algorithms with Markov chains", *Annals of Mathematics and Artificial Intelligence*, Vol. 5, p.79-88, 1991.

[5] A. Prügel-Bennett and J. L. Shapiro, An Analysis of Genetic Algorithms Using Statistical Mechanics, *Phys. Rev. Lett.* **72** (1994) 1305.

[6] M. Rattray, The Dynamics of a Genetic Algorithm under Stabilizing Selection, *Complex Systems* **9** (1995) 213.

[7] A. Prügel-Bennett and J. L. Shapiro, The Dynamics of a Genetic Algorithm for Simple Random Ising Systems, *Physica* **D104** (1997) 75.

[8] E. van Nimwegen, J. P. Crutchfield, M. Mitchell, Finite Populations Induce Metastability in Evolutionary Search *Phys. Lett.* **A 229** (1997) 144.

[9] Y. Fujiwara, technical report (unpublised).

[10] A. Prügel-Bennett, private communication.

[11] H. Kita, On Maintenance of Diversity in Genetic Algorithms, SICE (SY009/96/0000-0001), 1996.

[12] C. Adami, *Introduction to Artificial Life*, Springer-Verlag, 1998.

[13] K. Kaneko and T. Ikegami, *Fukuzatsukei no Shinka-teki Shinario*, Asakura, 1998 (Japanese).

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Toward the Realization of an Evolving Ecosystem on Cellular Automata

Hiroki Sayama
Department of Information Science, Graduate School of Science, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

Abstract

We have contrived the *evoloop*[5, 6], a new self-reproducing loop spontaneously evolving on a simple deterministic 9-state 5-neighbor cellular automata (CA) space. In this article, we examine the evolvability and adaptability of the *evoloop* through several experiments, the result of which brings us good prospects for the future implementation of an extraordinary large-scale artificial ecosystem on a superparallel machine environment.

Keywords

evolution, cellular automata, self-reproducing loop, structurally dissolvable self-reproducing loop

1 Introduction

The motivation of this study is to frame the basis for the realization of extremely large-scale evolutionary systems by means of a set of interacting *virtual state machines* embedded in a cellular automata (CA) space[2].

CA is a fine-grained parallel computational model intrinsically scalable and suitable for linear enlargement of simulation size, especially by implementing it on hardware[7, 8]. If we can create artificial evolutionary systems on CA, it will be very significant for the rapid enhancement of simulation scale, and such a great leap in expanding scale will be one of the most important factors for the reduction of gaps between artificial and real life.

Based on this idea, we have been trying to create evolving organisms on CA, and have proposed two improvements of Langton's famous self-reproducing loop[1]. The first improvement we made was the structurally dissolvable self-reproducing (SDSR) loop[4] which was capable of structural dissolution (a form of death) as well as self-reproduction. The second is the evolving SDSR loop, named *evoloop*[5, 6]. Its behavior being introduced in the next section suggests that it is relatively easy to realize evolutionary systems in a simple deterministic CA space.

In this article, we introduce the *evoloop* as the newest product of our study, and examine the evolvability and adaptability of it through several experiments. The result gives us good prospects for the future implementation of an extraordinary large-scale artificial ecosystem on a fine-grained superparallel machine environment with very simple algorithms inherited from the *evoloop*.

2 Evoloop: an evolving SDSR loop

The *evoloop* we have proposed is a new self-reproducing loop model on a simple deterministic 9-state 5-neighbor CA, constructed by enhancing robustness of the state-transition rules of the SDSR loop and slightly modifying its initial configuration. It actually has the ability to spontaneously evolve through the variation caused by direct interaction of phenotypes. Since the details of the *evoloop* will be carried on other literature[5, 6], they are omitted here. Its complete state-transition rule set is available from the author's WWW page introduced later.

Figure 1 shows a typical example of evolution of the *evoloop*. When the simulation begins, an ancestral loop soon proliferate all over the space. Then, self-reproduction and structural dissolution of loops begin to happen frequently in the space, which produce various kinds of variants. A self-reproducing loop of smaller species also emerges by accident from this melee, and once it appears, it is naturally selected due to its quick self-reproductive ability. Such an evolutionary process develops in the space as time proceeds, and eventually, the whole space gets filled with the loops of species 4 which is the strongest one in this world. It is quite characteristic of this evolutionary process that the variation in this world occurs first on phenotype (not on genotype) of the offspring being produced, by direct interaction of phenotypes, and then it consequently leads to alteration of genotype.

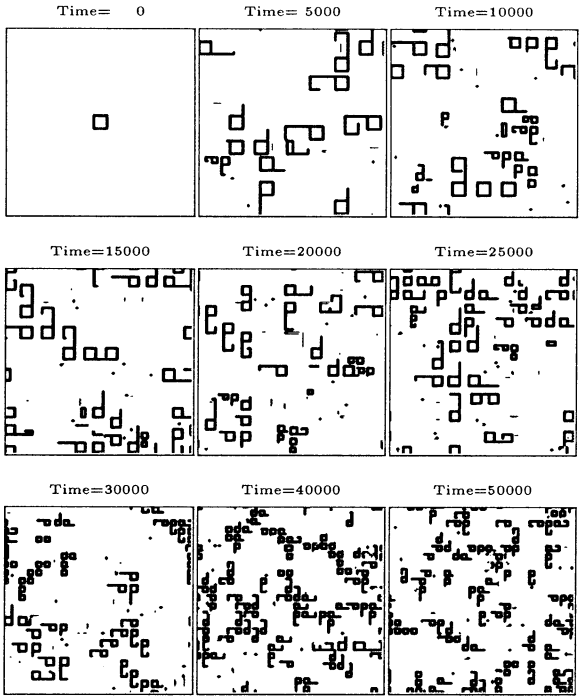


Figure 1: Temporal development of spatial configuration in a evolutionary process of *evoloops*. The ancestor is of species 13. The space is of 200 × 200 sites with periodic boundary conditions. As time proceeds, the smaller loops emerge and get superior to larger ones. The whole population gradually evolves toward smaller species, and finally the space gets filled with the loops of the strongest species 4.

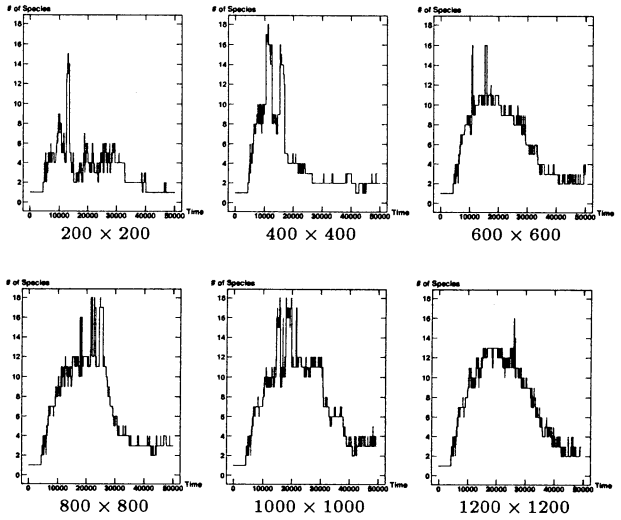


Figure 2: Temporal development of the diversity (the number of species) of *evoloops* in several cases with different sizes of space. Figures placed under each graph represent the size of space in that case. It is observed with respect to the cases with spaces smaller than 1000 × 1000 that, the larger the size of space becomes, the more easily various species can appear and co-exist with each other due to larger capacity of space.

3 Evolutionary/adaptive behaviors of the evoloop

In this section, we investigate the evolvability and adaptability of the *evoloop* from a viewpoint anticipating the future implementation on hardware.

3.1 Relation between the resource size and the evolvability

We expect that the size of the artificial world being simulated can be enlarged greatly by implementing it on hardware. The basic reason why we hope it is that we believe the larger the scale of simulation becomes, the more complexly and diversely the artificial system evolves. Here, we first examine this belief within the *evoloop*'s world.

In order to investigate the relation between the size of available resource and the evolvability of loops, the

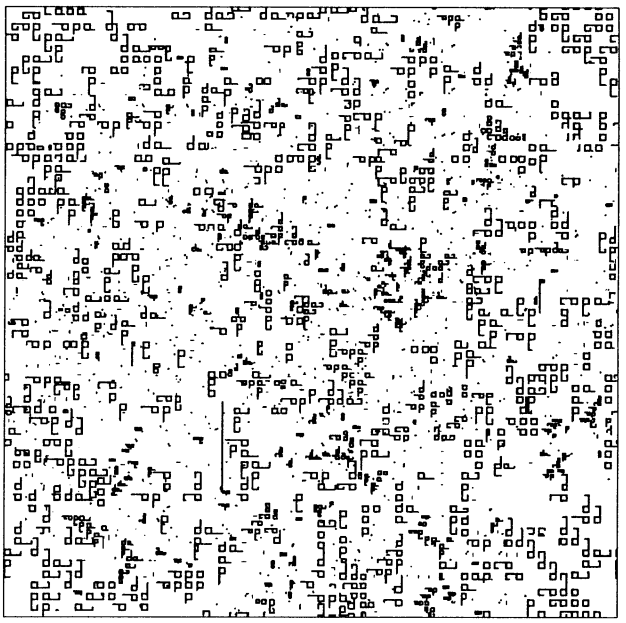


Figure 3: A snapshot taken in the evolutionary process of *evoloops* in the space of 1500 × 1500 sites (Time=20000). The space is islanded with the colonies of loops of various species.

evoloop was bred up in the spaces of various sizes ranging from 200 × 200 to 2000 × 2000 sites. Figure 2 indicates temporal development of the diversity (the number of species living in the space) of *evoloops* in several cases with different sizes of space. Since the number of possible species of the *evoloop* is limited, the diversity is saturated around the case of 1000 × 1000 sites. Nonetheless, it is possible to say with respect to relatively small spaces before the saturation that, the larger the size of space becomes, the more easily various species can appear and co-exist with each other (for a while) due to larger capacity of space. Figure 3 is an example of spatial configuration in such a state of diverse evolution emerging in a large space.

These results support the above-mentioned belief that large resource affords large diversity of evolution, and also suggests that, when we design much more complex organisms (like *Tierran*[3] creatures) on CA in the future, we need much larger resource in order to keep their diversity.

3.2 Emergence of self-reproducing organisms

We can observe the emergence of self-reproducing organisms from empty space by introducing a few additional state-transition rules and embedding small devices into space. The device used here is a non-deterministic finite-state automaton as small as a single site of the CA, and behaves stochastically to generate possible signal sequences meaningful in the *evoloop*'s world. Figure 4 depicts the state-transition diagram of this device. It generates random signal sequences filled with a mixture of a signal '7' (which means straight growth of the construction arm of the loop) and a pair of signal '4's (which mean left turning of the arm). Figure 5 shows a typical example of the emergence of self-reproducing organisms realized by using this device.

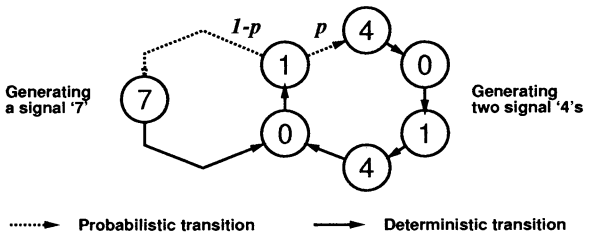


Figure 4: A state-transition diagram of the stochastic device used here. Probability p is set to be 0.05 in this study.

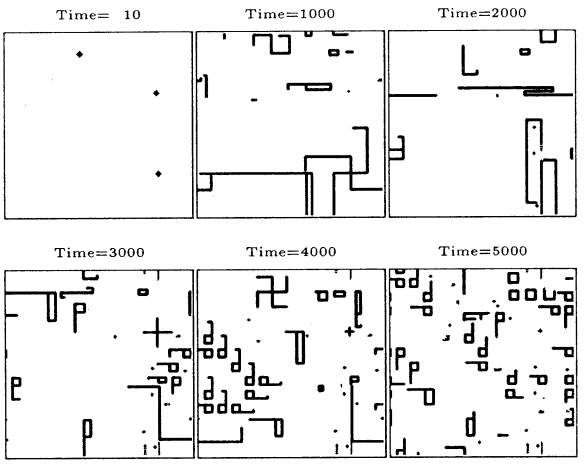


Figure 5: An example of the emergence of self-reproducing organisms from empty space. Three stochastic devices are randomly distributed in the space of 200 × 200 sites. These devices generate random signal sequences which cause the growth of pre-biotic arms. Self-reproducing organisms can emerge by accident from the interaction of these arms, and gradually proliferate over the space.

This phenomenon may be applicable for searching for configuration of viable organisms in an artificial world where only the grammer of genotypic signal sequence is known. This kind of method to embed small additional devices in a uniform array of machine modules would be easier and more practical if they were implemented on hardware.

3.3 Fault tolerance

Fault tolerance is an important feature of the model when its possibility to be implemented on hardware is considered. The more tolerant of faults the model becomes, the lower the level required for the hardware performance becomes, accordingly the easier the enhancement of circuit integration gets.

Here is the fault tolerance of the *evoloop* measured. Two kinds of faults are supposed in this measurement: the errors of state-transition rules which simulates unstableness of hardware modules, and the distribution of state-fixed blocks which simulates breakdown of hardware modules. The probabilities that the self-reproductive activity of *evoloops* will continue for 10000 updates are measured with the rates of the above two faults being changed.

The result is indicated in Table 1. It is understood from this table that the *evoloop* is actually tolerant of faults to some extent. The main reason of it is that the

Table 1: Probabilities that the self-reproductive activity of *evoloops* will continue for 10000 updates. The probability in each case was calculated based on the results of ten simulations of breeding *evoloops* of species 10 in the space of 200 × 200 sites with different seed values of random numbers. Emphasis is added on the probabilities not less than 0.5.

		Rate of errors of rules ($\times 10^{-6}$)				
		0.0	2.0	4.0	6.0	8.0
Rate	0.0	**1.0**	**0.8**	**0.8**	0.3	0.0
of	0.5	**1.0**	**0.5**	0.3	0.1	0.0
fixed	1.0	**0.7**	0.3	0.1	0.0	0.0
blocks	1.5	0.2	0.2	0.1	0.0	0.0
($\times 10^{-3}$)	2.0	0.1	0.1	0.0	0.0	0.0

evoloop can eliminate irregular configurations caused by faults quickly from the space, owing to its capability of structural dissolution. This feature may be regarded as quite favorable for hardware implementation. It is expected that the *evoloop* will be more tolerant of faults with further modification of refining the state-transition rules to be simpler and more robust.

4 Summary and future work

In this article, we introduced the *evoloop* spontaneously evolving in a simple CA space, and showed a couple of its features suitable for the future implementation on hardware: the increase of evolutionary diversity according to the increase of resourse size, the ability to autonomously search for self-reproductive configuration, and the fault tolerance. Of course, the model in this stage is too simple to produce enough complexity compared to real life; it still has plenty of room for improvement. However, we believe that it will be possible in the near future to synthesize an evolving ecosystem of unprecedented scale on super-parallel machine modules by utilizing the basic concept of the *evoloop* and refining the details of its model.

Finally, we should note here that a serious problem remains being unsolved, that is, the difficulty of observation of the behavior of evolutionary systems in CA space. Generally, meaningful configurations in CA are merely spatial collections of local states, thus all the judgments—which configuration is living or dead, how the genealogy of organisms is formed, when and where the variation occurs, e.g.—are left to the observer. This means that the observation of such a world needs enormous computation for image processing in order to extract significant information from the temporal development of configuration of the whole CA space. As the image processing also is a good target for the application of CA, we are planning to solve this problem by utilizing another CA specialized for image processing.

A WWW page which provides information about the *evoloop* is open at the URL below:
`http://proton.is.s.u-tokyo.ac.jp/~sayama/sdsr/`

Acknowledgments

We are grateful to Chris Langton, Yoshio Oyanagi, Yasuhiro Suzuki and Mari Sayama for their valuable help and advice in proceeding with this study.

References

1. Langton CG (1984), Self-reproduction in cellular automata. *Physica D* 10:135–144.
2. Langton CG (1986), Studying artificial life with cellular automata. *Physica D* 22:120–149.
3. Ray TS (1990), An approach to the synthesis of life. In: Langton CG, Taylor C, Farmer JD, Rasmussen S (eds), *Artificial Life II: Proceedings of the Workshop on Artificial Life*, Santa Fe, NM, Feb 1990, pp.371–408.
4. Sayama H (1998), Introduction of structural dissolution into Langton's self-reproducing loop. In: Adami C, Belew RK, Kitano H, Taylor CE (eds), *Artificial Life VI: Proceedings of the Sixth International Conference on Artificial Life*, Los Angeles, CA, Jun 1998, pp.114–122.
5. Sayama H (1998), Spontaneous evolution of self-reproducing loops implemented on cellular automata: A preliminary report. In: Bar-Yam Y (ed), *Proceedings of the Second International Conference on Complex Systems*, Nashua, NH, Oct 1998, forthcoming.
6. Sayama H, A new structurally dissolvable self-reproducing loop evolving in a simple cellular automata space. Submitted.
7. Sipper M, Mange D, Stauffer A (1997), Ontogenetic hardware. *BioSystems* 44:193–207.
8. Toffoli T, Margolus N (1987), *Cellular Automata Machines*. MIT Press, Cambridge, MA.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Prediction of Deviant Genetic Codes — Why They Evolve —

Tetsuya Maeshiro

Dept.6, ATR Human Information Processing Labs.,
2-2 Hikaridai, Seika, Soraku, Kyoto, 619-0288 JAPAN.

Abstract

All currently known deviant genetic codes are predicted based on the assumption that the changeability and robustness, two paradoxical but fundamental properties for the survival and evolution of organisms, are the selection pressure on the evolution of deviant genetic codes from the standard genetic code. These measures are defined as the intrinsic properties of genetic codes. Codon reassignments found in known deviant codes generally give the highest improvements on robustness and changeability, and new deviant genetic codes discovered in future may belong the the set of predicted deviant codes.

1 Introduction

The hypothesis that requests for both robustness and changeability have a strong influence on the origin of the standard genetic code (SGC) (Table 1) and its evolution to deviant codes (Table 2) explains consistently the structural regularity of genetic codes and why SGC evolved to many deviant codes[1]. The robustness and changeability are paradoxical properties, and they are respectively related with the ability of organisms to survive and evolve. The deviant genetic codes are so called because some codons are reassigned to phenotypes different from those of SGC.

Jukes and Osawa[2] proposed the biased codon usage as a mechanism to generate deviances in genetic codes. No plausible hypothesis, however, was given to explain why such deviances originated. Evidently any change in a component with importance of genetic code results in drastic changes of organisms, with high probability of death [extinction]. Consequently, it seems plausible to assume that changes in genetic codes are rare events. On the other hand, same deviant genetic code appeared independently and multiple times at least in ciliates[3], and same deviant genetic code is found in mitochondrial genetic systems of nematoda and arthropoda (Table 2), whose common ancestor is almost the origin of animals, suggesting their independent origin. These facts indicate that the codon reassignment is a nonrandom process. This paper presents a possible selection pressure on the evolution of genetic codes, which limits the freedom of codon reassignments and predicts all currently known deviant genetic codes.

The robustness is defined by two properties: the μ-robustness, which is the unalterability of phenotypes due to a single base mutation of codons, where the phenotypes denote any of twenty amino acids and the stop codon; and the s-robustness, which is the robustness against nonsense mutations. The changeability is the alterability of phenotypes by a single base mutation of codons.

2 Robustness and Changeability of Genetic Codes

The genetic code is a coding table between 64 codons and 21 phenotypes. The robustness and changeability of genetic codes are calculated based on their graph representation. A node represents a set of codons assigned to the same phenotype, where any pair of codons in the set can be changed by a successive mutation in a single base, Therefore, the six codons coding Ser is divided into two nodes. Two nodes are connected by an edge when at least one codon in a node differs only in single base from a codon in another node.

The μ-robustness is the average probability that a codon keeps coding the same phenotype when a single base of codon mutates.

An opposite property to μ-robustness, the changeability measures the average of the transition probabilities along the shortest paths between all of the pairs of phenotypes in the graph representation of the code, because the shortest

Table 1: Standard Genetic Code

1st base	2nd base U	2nd base C	2nd base A	2nd base G	3rd base
U	UUU Phe	UCU Ser	UAU Tyr	UGU Cys	U
	UUC Phe	UCC Ser	UAC Tyr	UGC Cys	C
	UUA Leu	UCA Ser	UAA *stop*	UGA *stop*	A
	UUG Leu	UCG Ser	UAG *stop*	UGG Trp	G
C	CUU Leu	CCU Pro	CAU His	CGU Arg	U
	CUC Leu	CCC Pro	CAC His	CGC Arg	C
	CUA Leu	CCA Pro	CAA Gln	CGA Arg	A
	CUG Leu	CCG Pro	CAG Gln	CGG Arg	G
A	AUU Ile	ACU Thr	AAU Asn	AGU Ser	U
	AUC Ile	ACC Thr	AAC Asn	AGC Ser	C
	AUA Ile	ACA Thr	AAA Lys	AGA Arg	A
	AUG Met	ACG Thr	AAG Lys	AGG Arg	G
G	GUU Val	GCU Ala	GAU Asp	GGU Gly	U
	GUC Val	GCC Ala	GAC Asp	GGC Gly	C
	GUA Val	GCA Ala	GAA Glu	GGA Gly	A
	GUG Val	GCG Ala	GAG Glu	GGG Gly	G

paths between the nodes practically determine the transition probabilities, and consequently, the changeability of the code. For paths between nodes of amino acids, those paths linked by the node of stop codons are removed, denoted interrupted paths, because they correspond to the nonsense mutations that result in the synthesis of shorter proteins, and most of them have no biological activity.

The third property, s-robustness, measures the robustness against nonsense mutations, considers the interrupted paths that were excluded from the calculation of the changeability due to the deleterious consequences of nonsense mutations.

The μ-robustness, s-robustness, and changeability of genetic codes become relevant when the DNA sequence changes, particularly through replication. These measures are related with the survivability and adaptability of species. With a high μ-robustness, the probability to conserve the protein sequence and its functionality is high. On the other hand, a high changeability gives larger variations of amino acid sequences after replications.

The s-robustness is related to both the robustness and changeability, and measures the probability of nonsense mutations when an amino acid mutates into another amino acid. Genetic codes with a high s-robustness allow mutations between amino acids with a low probability of nonsense mutations when two or more single base mutations are necessary.

3 Prediction of Deviant Genetic Codes

We assume that one deviance is introduced at a time. A hypothetical anticodon list of SGC is used, where each tRNA recognizes exactly one codon to model the change in the anticodon list, simulating the change in an amino acid associated with a tRNA and the appearance/disappearance of a tRNA. The following procedure is used to predict Deviant genetic codes, simulating the evolution of deviant genetic codes from SGC.

1. Start from SGC.
2. Generate all possible genetic codes introducing one deviance. Genetic codes not coding all 21 phenotypes are rejected.
3. Select generated genetic codes with improvement in changeability, μ-robustness or s-robustness.
4. Classify the generated genetic codes according to the improvement manner of three measures, for instance, improvement

only in changeability, and improvement in all three.

5. For each generated genetic code, go to step 2.

There are three types of SGC according to the anticodon list[5], and appropriate anticodon list of SGC can be used to increase the precision of prediction, which predicts the reassignment of multiple codons.

Figure 1 shows the optimality of known deviant genetic codes in the set of predicted genetic codes, represented as a phylogenetic tree. The hypothesis on robustness and changeability results in a small number of allowed codon reassignments compared to the total number of possible reassignments. Furthermore, the reassignments of existing deviant codes generally gives the highest improvement when the robustness is improved, but the optimality is not high when the changeability is improved. This is because the changeability is a global property of genetic codes, so the number of possible reassignments to improve the changeability is higher than that of robustness, which is a local property.

The position to assign the 21st amino acid, selenocysteine (Sec), present in some organisms[4], is also predicted. Its optimality is the second among 17 valid reassignments, from the total of 62 possible reassignments.

Figure also shows the predicted order of codon reassignments, which agrees with the manually predicted order based on the phylogenetic tree of species with deviant genetic codes. The manual prediction is based on the assumption that the common deviances appeared first, and particular deviances in the later evolutionary stage.

4 Conclusions

Our hypothesis suggests that the robustness and changeability have strongly influenced on the evolution of deviant genetic codes. Since the structural regularity of SGC is also explained[1], our model provides a plausible explanation on the structure of SGC and its evolution to deviant codes, and offers a theoretical basis for understanding an important role of genetic codes, which is to determine how the mutation in DNA sequence is reflected in amino acid sequences, and consequently on the functionality of organisms.

The hypothesis of robustness and changeability gives most probable codon reassignments, and it explains the presence of similar deviant genetic codes in phylogenetically distant species. Certainly physico-chemical factors have influenced the evolution of deviant codes, but the prediction of deviant codes suggests that the robustness and changeability are also important. New deviant codes have been discovered every one or two years, and if new deviant codes are discovered, they probably belong to the set of predicted codes.

Acknowledgments I thank K. Shimohara for support.

1. Maeshiro T, Kimura M (1998), The role of robustness and changeability on the origin and evolution of genetic codes. Proc. Natl. Acad. Sci. USA 95:5088-5093.

2. Osawa S, Jukes TH, Watanabe K, Muto A (1992), Recent evidence for evolution of the genetic code, Microbiol. Rev., 56:229-264.

3. Tourancheau AB, Tsao N, Klobutcher LA, Pearlman RE, Adoutte A (1995) Genetic code deviations in the ciliates: evidence for multiple and independent events, EMBO J., 14:3262-3267.

4. Zinoni F, Birkmann A, Stadtman TC, Böck A (1986) Nucleotide sequence and expression of selenocysteine-containing polypeptide of formate dehydrogenase (formate-hydrogen-lyase linked) from *Escherichia coli*, Proc. Natl. Acad. Sci. USA, 83:4650-4654.

5. Osawa S, Jukes TH (1989) Evolution of the genetic codes as affected by anticodon content, TIG, 4:191-198.

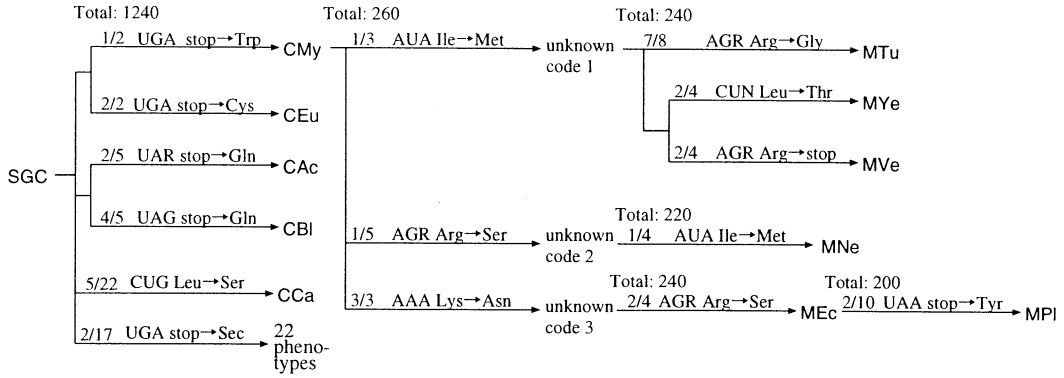


Figure 1: Deviant codes predicted from SGC represented as an evolutionary tree. Only the codon reassignments found in known deviant codes are shown. SGC with the mitochondrial anticodon list was used to predict mitochondrial codes. The numbers labeled with "Total" indicate the total number of codes with 21 phenotypes. The edges are labeled with the number of predicted codes with a similar manner of improvement, and the optimality of the reassignment.

Table 2: Assignments of deviant codons. The abbreviations denoted in "Code" are used. `N` denotes any of `A`, `U`, `G`, and `C`, and `R` denotes `A` and `G`. Compiled from http://www3.ncbi.nlm.nih.gov/htbin-post/Taxonomy/wprintgc?mode=c

Representative Genetic System		Code	Changes from SGC	
			codon	phenotype
Mitochondrial	Yeasts	MYe	`UGA`	stop ⇒ Trp
			`AUA`	Ile ⇒ Met
			`CUN`	Leu ⇒ Thr
Mitochondrial	Platyhelminths	MPl	`UGA`	stop ⇒ Trp
			`AAA`	Lys ⇒ Asn
			`AGR`	Arg ⇒ Ser
			`UAA`	stop ⇒ Tyr
Mitochondrial	Nematoda	MNe	`UGA`	stop ⇒ Trp
	Arthropoda		`AGR`	Arg ⇒ Ser
	Mollusca		`AUA`	Ile ⇒ Met
Mitochondrial	Echinodermata	MEc	`UGA`	stop ⇒ Trp
			`AAA`	Lys ⇒ Asn
			`AGR`	Arg ⇒ Ser
Mitochondrial	Tunicata	MTu	`UGA`	stop ⇒ Trp
			`AUA`	Ile ⇒ Met
			`AGR`	Arg ⇒ Gly
Mitochondrial	Vertebrata	MVe	`UGA`	stop ⇒ Trp
			`AUA`	Ile ⇒ Met
			`AGR`	Arg ⇒ stop
Mitochondrial	Euascomycetes	MEu	`UGA`	stop ⇒ Trp
Nuclear	Mycoplasma	CMy		
Nuclear	Euplotes	CEu	`UGA`	stop ⇒ Cys
Nuclear	Acetabularia	CAc	`UAR`	stop ⇒ Gln
Nuclear	Blepharisma	CBl	`UAG`	stop ⇒ Gln
Nuclear	Candida	CCa	`CUG`	Leu ⇒ Ser

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

On a correlation between the degree of halting property and the qualitative behavior of Abstract Chemical System

Yasuhiro Suzuki, and Hiroshi Tanaka
Medical Research Institute, Tokyo Medical and Dental University
1-5-45 Yushima, Bunkyo-ku Tokyo Japan
Email: {suzuki.com@.tmd.ac.jp, tanaka@tmd.ac.jp }

Abstract

We develop an abstract computational model (ARMS), which can deal with systems with many degrees of freedom and confirm that it can simulate the emergence of complex cycles such as chemical oscillations that are often found in the emergence of life. We also study mathematical properties of the model by using a computational algebra and propose an order parameter to describe the global behavior of the system.

Keywords: Abstract Chemical System, Chemical oscillation, Emergent computation.

1 Model

In this section, we describe an abstract chemical system in terms of an abstract rewriting system. Before describing the system in detail, we introduce abstract rewriting systems in general.

Abstract Rewriting System (ARS) An abstract rewriting system models the algebraic characteristics of calculation. The principle of calculation within an ARS is simple. A calculation is performed by rewriting using rules as in formal grammar: $a \to Sa$.

1.1 ARMS

Extending the concepts of the abstract rewriting system, we introduce an abstract rewriting system on multi-sets (ARMS) [4]. Intuitively, ARMS is like a chemical solution in which floating *molecules* can interact with each other according to reaction rules. Technically, a chemical solution is a finite multi-set of elements denoted $A^k = \{a, b, \ldots, \}$; these elements correspond to *molecules*, and reaction rules are specified in terms of rewrite rules. As to the intuitive meaning of an ARMS, we refer to the study of chemical abstract machines [2].

We denote the empty set by ϕ, and the *base number* of a multi-set (size of a multi-set) by $|S|$ (where S is a multi-set), respectively. The multi-sets correspond to possible states of *chemical solution*. The set of multi-sets corresponds to the space of transitions of an ARMS.

Definition 1 (Rewriting rule) *A "rewriting rule" is a relation $l\ \mathcal{R}\ r$ ($l.\ r \in \Sigma$). $|l|, |r| \leq$ maximal multi-set size, n. A rewriting rule $l\ \mathcal{R}\ r$ is denoted as $l \to r$.*

A rewriting rule such as

$$a \to a \ldots b. \tag{1}$$

is called a *heating rule* and denoted as $r_{\Delta>0}$; it is intended to contribute to the stirring solution. It breaks a complex *molecule* into smaller ones: *ions*. On the other hand, a rule such as

$$a \ldots c \to b. \tag{2}$$

is called a *cooling rule* and denoted as $r_{\Delta<0}$; it rebuilds *molecules* from smaller ones. In this paper, reversible reactions, i.e., $S \rightleftharpoons T$, are not considered. We shall not formally introduce the refinement of *ions* and *molecules* though we use refinement informally to help intuition.

Definition 2 (ARMS) *An "Abstract Rewriting System on Multi-sets" (ARMS) is a pair (T, Ru) consisting of a multi-set T and a set Ru of rewriting rules.*

1.2 How ARMS works

In ARMS, we assume that one randomly selected rule is applied in each rewriting step, unless no input is allowed.

Example In this example, we assume that a will be inputted on each rewriting step, the maximal multi-set size is 4 and the initial state is given by $\{a, a, f, a\}$. The set of the rewriting rules, Ru_1 is $\{r_1, r_2, r_3, r_4\}$, where each rule is described by the following:

$$aaa \to b : r_1,\ b \to a : r_2,\ b \to c : r_3,\ a \to bb : r_4.$$

In this example, we assume that rules are selected as following the order $\{r_4 \Rightarrow r_1 \Rightarrow r_3 \Rightarrow r_2\}$. Then, each rule is applied in the following way. First, r_4 is applied. Next, as steps 2 and 3, r_1 and r_3 are applied, respectively. Finally, as step 4, r_2 is applied.

$\{aafa\}$ $\subseteq a$ (the left hand side of r_4)
$\downarrow$ .... can not input a and can not apply r_4,
$\{aafa\}$ $\subseteq aaa$ (the left hand side of r_1)
$\downarrow$ .... can not input a but can apply r_1
$\{bf\}$

Figure 1: Example of rewriting steps of ARMS

Figure 1 illustrates two rewriting steps of the calculation from the initial state.

As the first step, since the base number of the multi-set is 4, the system can not input a. On the left hand side of r_4, a is included in $\{aafa\}$, however, r_4 can not be used. If a is replaced with bb, the base number of the multi-set becomes 5 and it exceeds the maximal multi-set size, 4.

In the next step, the system can not input a, however, r_1 can apply to the multi-set and $\{aafa\}$ is rewritten into $\{bf\}$ (because if aaa is replaced with b, the base number of the multi-set does not exceed the maximal multi-set size, see Figure 1).

Experimental Results of the Simulation of ARMS We simulated ARMS with various different setups; in this paper we shall discuss two of them as follows:

- Simple setup
- Brusselator model.

Through these experiments, we confirmed that the system is capable of generating complex patterns [4].

2 Order parameter for ARMS

From an examination of the effectiveness of the termination property, we obtained the λ_e parameter as an order parameter for the qualitative behavior of ARMS.

We define "order" here as being given by the *diversity* of cycles. Thus, in this paper, "ordered state" refers to a case where the system yields simple cycles (such as the limit cycle), while "disordered state" refers to the case where the system yields chaotic or complex cycles.

2.1 The λ_e parameter

Let us define the λ_e parameter as follows:

$$\lambda_e = \frac{\Sigma r_{\Delta S>0}}{1 + (\Sigma r_{\Delta S<0} - 1)} \tag{3}$$

where $\Sigma r_{\Delta S>0}$ corresponds to the number of *heating rules* used, and $\Sigma r_{\Delta S<0}$ to the number of *cooling rules* used. This parameter is defined when the number of rules used is greater than 1.

When the ARMS only uses rules of the type $r_{\Delta S<0}$, λ_e is equal to 0.0. On the contrary, if the ARMS uses rules of the type $r_{\Delta S>0}$ and $r_{\Delta S<0}$ with the same frequency, λ_e is equal to 1.0. Finally, when the ARMS only uses rules of the type $r_{\Delta S>0}$, λ_e is greater than 1.0.

3 Simulation

We confirmed the appropriateness of the λ_e parameter through a simulation of the ARMS, and verified that the parameter reflects the diversity of cycles that are generated by the system.

Method We assume that the maximal multi-set size is 10. At the beginning of a simulation, the value of p is set to 0 and it increased by steps of 0.01. At each value of p, 100 new initial states with base number between 1 and 10 are generated by selecting the symbol a or b randomly. The base number of the initial state of a multi-set is decided randomly. For each initial state the simulation is performed for 1000 steps.

4 Experimental results

Let us present the experimental results, focusing on the following two points:

(1) the correlation between the system's terminating property and the value of p,

(2) the correlation between the diversity of periods of the generated cycles and the value of p.

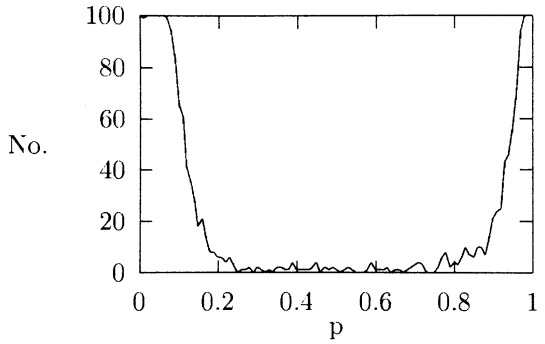


Figure 2: The correlation between the number of terminating calculations and p

Termination Figure 2 illustrates the correlation between p and the number of terminating calculations. In Figure 2, the vertical axis corresponds to the number of terminating calculations and the horizontal axis corresponds to p.

In this simulation, before p exceeded 0.1, most calculations terminated. When p exceeded 0.1, the number of terminating calculations decreased rapidly, while when p was greater than 0.2, this decrease leveled off. Then, for p between 0.3 and 0.85, only a few calculations terminated, while with p greater than 0.85, the number of terminating calculations increased rapidly again. In other words, for p near 0 or 1, the system strongly terminated, while with when p away from 0 or 1, the termination property became weak.

The number of generated cycles Figure 3 illustrates the relationship between the number of generated cycles and p.

As p increases, the number of generated cycles also increases rapidly. This rise levels off when p exceeds

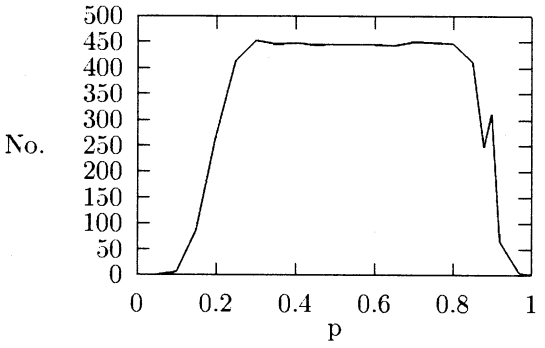


Figure 3: Correlation between the number of generated cycles (vertical axis) and p (horizontal axis).

0.3. For p between 0.3 and 0.8, the number of generated cycles remains at the same level, however, when p exceeds 0.75, this number rapidly decreased again.

This result indicates that the termination property indeed is related to the number of generated cycles. As we mentioned in the previous subsection, when p was close to 1 or 0, the system strongly terminated while for p away from 0 or 1, this property became weak. Also, for p near 1 or 0, only a few cycles were generated while for p away from 1 and 0, the system generated many cycles. We may thus conclude that the degree of termination influences the number of cycles generated by the system.

It is interesting that once the number of generated cycles reaches around 450, it remains at the same level, even while p was changed from 0.3 continuously up to 0.8. We believe that for p between 0.3 and 0.8, the system is in an equilibrium state.

To investigate the system's behavior in this equilibrium state, let us examine the relationship between the kinds of periods generated by the cycles and the value of p.

The kinds of periods generated by the cycles The experimental result indicates that even if the system is in an equilibrium state, the kinds of periods are different for each value of p. Figure 4 displays the

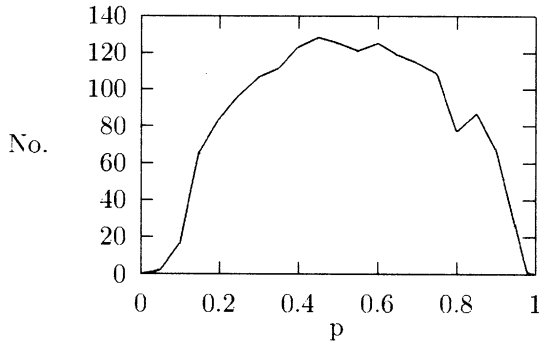


Figure 4: Correlation between the kinds of periods of generated cycles (vertical axis) and p (horizontal axis).

average number of different kinds of periods. As we can see in this figure, when p reaches about 0.5, the number of different kinds of periods is maximal. In other words, when *cooling rules* and *heating rules* are used at the same frequency, many kinds of periods are generated.

5 Discussion

In this section, we shall show that the λ_ϵ parameter is related to the termination property, which implies that Langton's λ parameter is also related to termination.

Wolfram's classification and Langton's λ parameter as qualitative characterizations of the behavior of cellular automata are well known.

Wolfram classes Wolfram ([3]) proposed four classes of qualitative behavior patterns of cellular automata based on his investigation of a large sample of CA rule tables. He maintained that any cellular automata would fall into one of the four basic classes': **Class 1**: Evolution leads to a homogeneous state, **Class 2**: Evolution leads to a set of separated simple stable or periodic structures, **Class 3**: Evolution leads to a chaotic pattern, and **Class 4**: Evolution leads to chaotic, localized structures.

Langton's λ parameter The λ parameter is defined as follows:

$$\lambda = \frac{K^N - n_q}{K^N}, \tag{4}$$

where K corresponds to the number of symbols used, N to the size of the neighborhood, and n_q to the number of local rules which transform cellular automata to the *quiescent* state, respectively. The quiescent state is picked arbitrarily, and is usually associated with a "special" state, such as the "zero" state.

5.1 Computational algebraic characteristics of λ parameter

We demonstrate here that the λ parameter indicates a degree of *termination* and that when cellular automata yield complex or chaotic behavior, this termination property of the system becomes weak. When λ is close to 0, the dynamical activity of the cellular automaton quickly dies out or reaches a uniform fixed point. In this regime, thus, the termination property of cellular automata is strong. As the value of the λ is increasing, cellular automata evolve to periodic structures or chaotic aperiodic patterns. As the value of the λ increases even more, the termination property becomes weak. The larger the parameter becomes, the later the calculation terminates. Since th transitional space of cellular automata is finite, if a calculation does not terminate or is difficult to stop, cyclic structure must emerge in the process of the computation.

termina -ting	λ_ϵ	λ	Wolfram classes
Strong	$\lambda_\epsilon \cong 0.0$, $\lambda_\epsilon \gg 1.0$	$\lambda \cong 0.0$	I or II
Fair strong	$1.0 > \lambda_\epsilon > 0.0$. $\lambda_\epsilon > 1.0$	$0.0 < \lambda < 1 - \frac{1}{K}$	II or IV
Weak	$\lambda_\epsilon \cong 1.0$	$\lambda \cong 1 - \frac{1}{K}$	III

Table 1: Relation among parameters, complexity classes

The qualitative dynamics of CA and ARMS that we have described is summarized in Table 2. It suggests that each parameter and class indicate a degree of terminating.

5.2 Comments on the 'edge of chaos'-regime

We have demonstrated that an equal frequency of $r_{\Delta S>0}$ rules to $r_{\Delta S<0}$ rules yields dynamical patterns for both cellular automata and ARMS. We can now see that the principle of "edge of chaos", according to the dynamics just described, results from a biased ratio of $r_{\Delta S>0}$ rules to $r_{\Delta S<0}$ rules.

If these two types of rules are used at the same frequency, the system yields a chaotic pattern. However, when the ratio changes slightly, the "edge of chaos" regime emerges. Thus, a lack of symmetry in the application of rules generates the diversity of cycles in these systems.

References

[1] Langton, C. G, Life at the edge of chaos. In *Artificial Life II*, edited by C. G. Langton, C. Taylor, J. D. Farmer, and S. Rasmussen. Redwood City, CA: Addison-Wesley., 1991.

[2] Bellin, G. and G. Boudol. The chemical abstract machine. *Theoretical Computer Science* 96: 217 248. 1992.

[3] S. Wolfram, Cellular Automata and Complexity, Addison Wesley, 1994.

[4] Y. Suzuki and H. Tanaka. Order Parameter for Symbolic Chemical System. pp 130-139. MIT press, 1998.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Control of a Humanoid Robot using a Multi-Freedoms Motion Capture Device

S. Kurono, Y. Miyamoto
Dept. of Electrical Engineering
Kyushu Sangyo University
2-3-1,Matsukadai,Higashi-ku,
Fukuoka-shi,813,Japan

S. Aramaki
Dept. of Electronics Engineering
Fukuoka University
8-19-1,Nanakuma,Jonan-ku,
Fukuoka-shi,814-01,Japan

Abstract

We have developed a small humanoid robot which has 20 degrees of freedom and continued a study in order to realize some complicated motion such as walking with two legs. But, it is very difficult to make a sequence of references in order to control a humanoid robot which has very large number of degrees of freedom. In our study, we developed a motion capture device which is able to input 20 degrees of joint angle in real time. We have been studying to realize walking motion of our humanoid robot by controlling the robot based on master/slave control method using the motion capture device. In this paper, we introduce the mechanical structure of it, calibration method to make use of it and outline of our experimental system.

1 Introduction

We developed a small humanoid robot which has 20 degrees of freedom and about 30Cm tall, and a multi-freedoms controller based on a FPGA chip which is able to control a robot having 24 degrees of freedom. We have realized the walking motion of this humanoid robot with the multi-freedoms controller mentioned above, and introduced the structure of the control system on the AROB 3rd '98 last year. Commonly, it is not easy to make a sequence of references in order to control a humanoid robot which has very large number of degrees of freedom. In our study, we developed a motion capture device which is able to input 20 degrees of joint angle in real time. Fig.1 shows the general view of the whole system in our study.

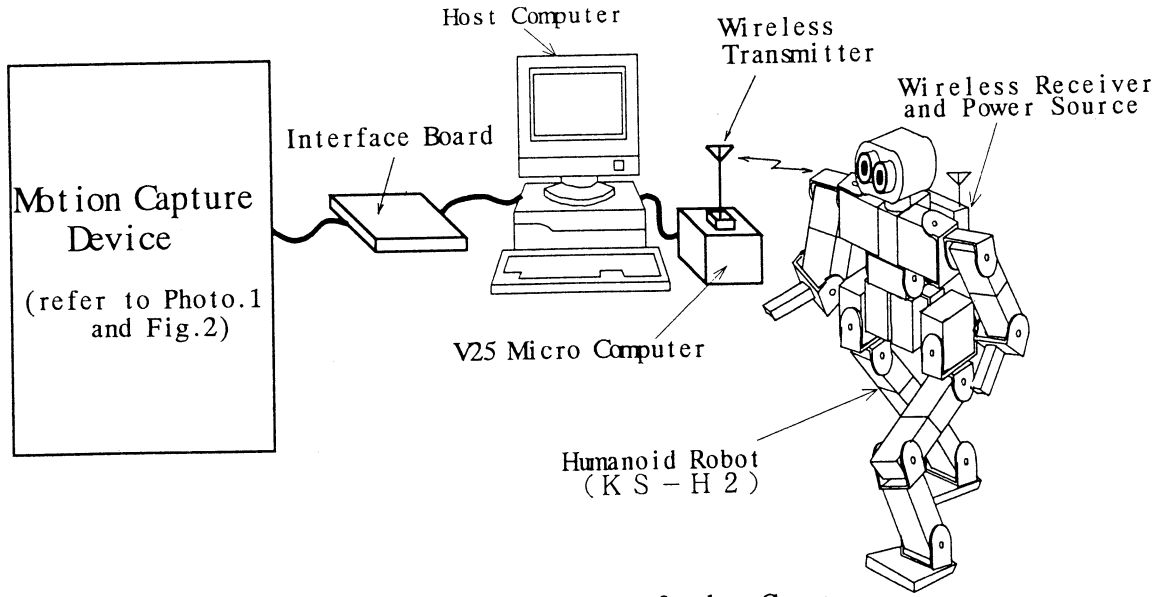
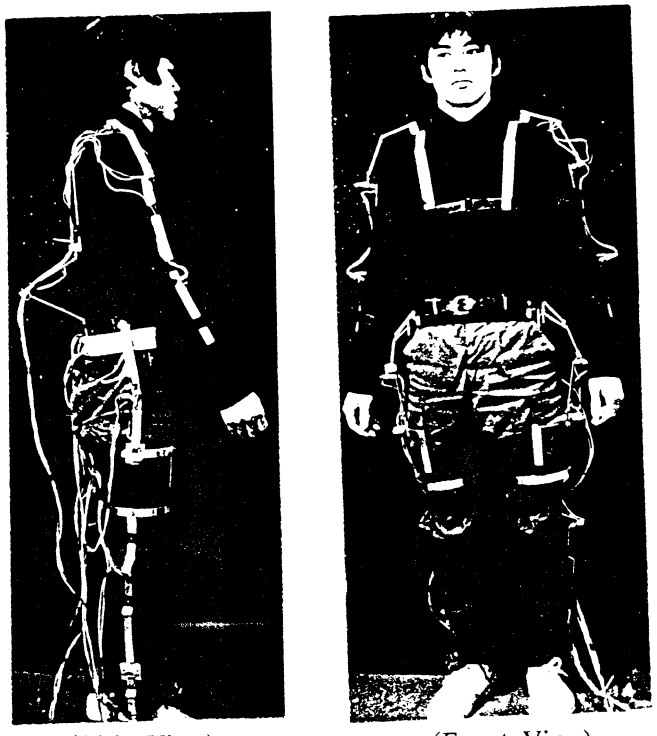


Fig.1 General view of the System

The system is composed with a multi-freedoms motion capture device, a host computer, wireless transmitter/receiver, a multi-freedoms controller and a humanoid robot. A wireless receiver, a multi-freedoms controller and a power source are mounted on the back of the humanoid robot.

2 Development of a multi-freedoms motion capture device

Photo.1 shows front and side view of the motion capture device mounted on a human body. Fig.2 shows a rough sketch of the device.



(Side View) (Front View)

Photo.1 Motion capture device mounted on a Human body

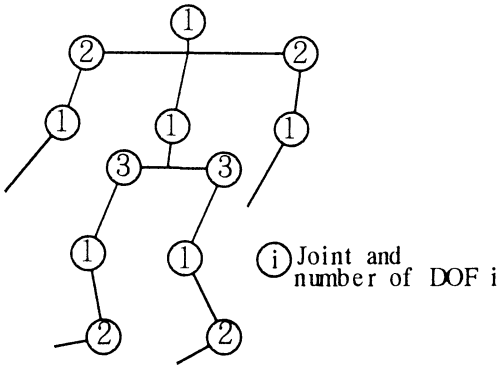


Fig.3 Structrue of DOFs of the motion capture device

The motion capture device posseses 20 joints corresponding to those of human's respectively. Rotations of each joint are sensed by each potentiometer equipped on it, and input to host computer through A/D converter in real time. The input data for each joint are simultaneously transmitted to the humanoid robot through wireless transmitter after some modifying processes. Fig.3 shows the structure of DOFs(Degrees Of Freedom) of the motion capture device. In the rest of this section, we illustrate the structure of sensors arranged on each parts of a human body from the upper parts of it. Fig.4 shows the arrangement for 2 DOFs of a shoulder. As shown in the figure, two hinges connecting links make the rotation of potentiometer smoother.

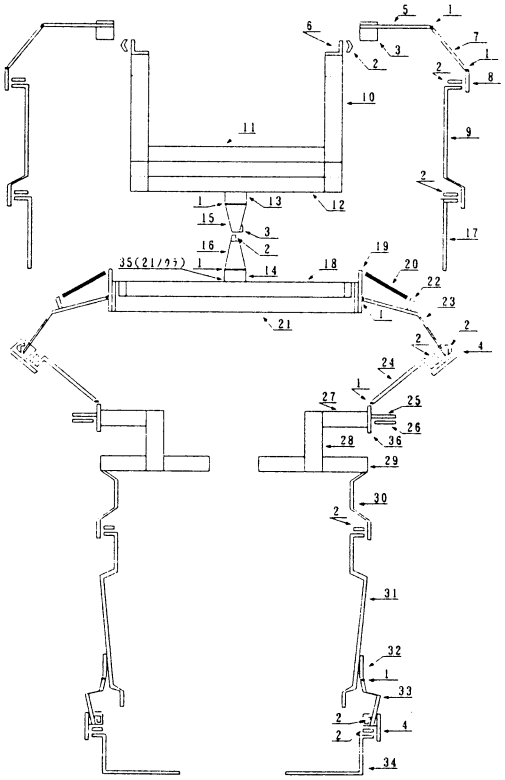


Fig.2 A sketch of the motion capture device

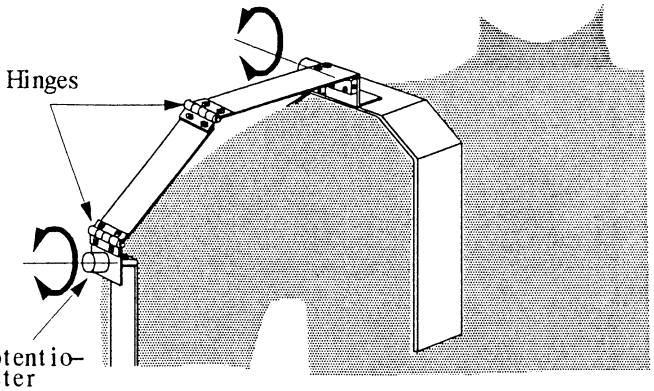


Fig.4 2 DOFs of a Shoulder

Fig.5 shows the sensor for detecting bending motion of an elbow and a knee. Fig.6 shows the arrangement for 2 DOFs of a thigh. Two springs are used to pull up the lower parts of motion capture device as shown in the figure. These springs prevent the lower arrangements from slipping down and make the rotations of potentiometers easier and smoother. Fig.7 shows the arrangement for detecting twisting motion of a leg. As shown in the figure, the twisting motion is detected by a wheel fixed to the axis of a potentiometer and a rubber band fixed on a leg. The link mounting the potentiometer is capable of sliding around the circular surface of a leg by making use of 4 pulleys arranged as in the figure. Fig.8 shows the arrangement for 2 DOFs of an ankle. The link fixed to the axis of a potentiometer is capable of sliding along the longitudinal direction of a leg. This arrangement makes the rotation of a potentiometer much smoother as easily understandable by taking a look at the figure. Fig.9 shows the arrangement for detecting the bending motion of the waist. This is an easy and effective method to detect bending motion of the waist.

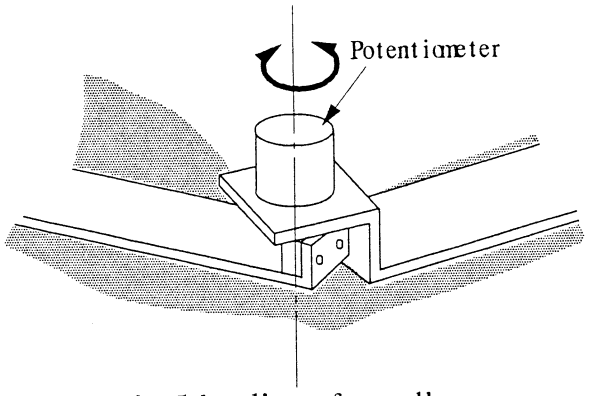


Fig.5 bending of an elbow and a knee

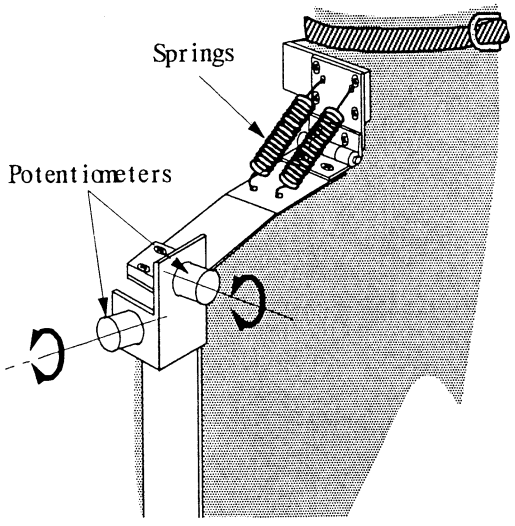


Fig.6 2 DOFs of a Thigh

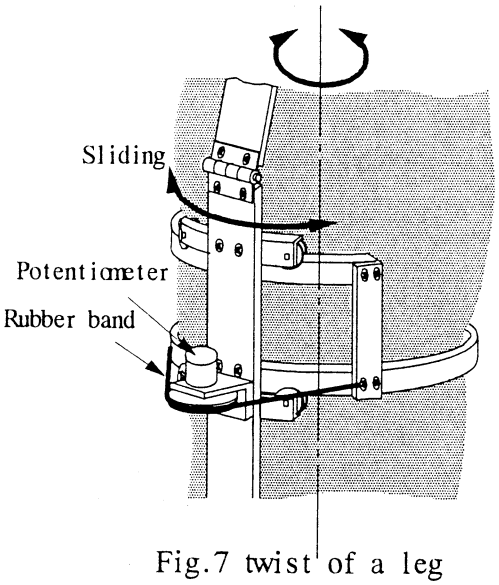


Fig.7 twist of a leg

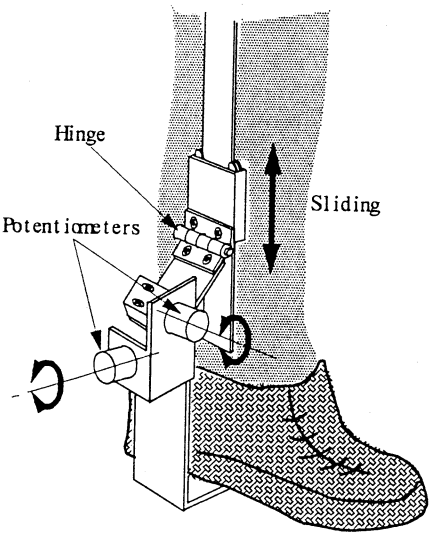


Fig.8 2 DOFs of an ankle

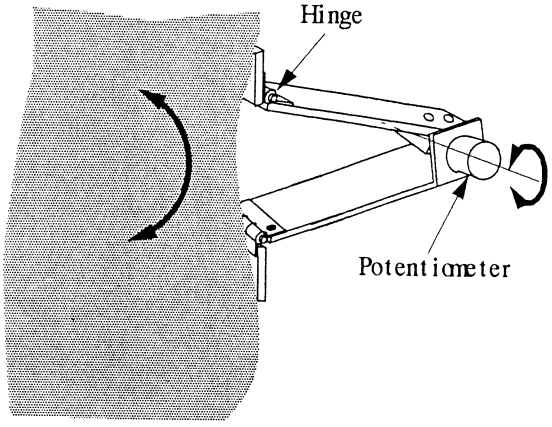


Fig.9 bending of the waist

3 Calibration of the motion capture device

In order to use the motion capture device in practice, it is required to get the relationship between references to serbomechanisms r_i and analog inputs a_i for all DOFs ($i = 1 \sim 20$) as shown in Fig.10. Fig.11 shows several examples of postures to get a pair of coordinates in Fig.10, (a_{i0}, r_{i0}) and (a_{i1}, r_{i1}). After the relationship between references r_i and analog inputs a_i for all DOFs is given, r_i is calculated from a_i by the next equation.

$$r_i = \frac{r_{i1} - r_{i0}}{a_{i1} - a_{i0}}(a_i - a_{i0}) + r_{i0}$$

In the equation above, the slopes of lines $K_i = \frac{r_{i1} - r_{i0}}{a_{i1} - a_{i0}}$ are considered to be invariable for each joints. Therefore, once $K_i (i = 1 \sim 20)$ are given, references r_i can be calculated by the next euation.

$$r_i = K_i(a_i - a_{iN}) + r_{iN}$$

Here, the coordinates (a_{iN}, r_{iN}) are corresponding to neutral points somewhere within the rotations of joints. This means that calibration work is made much easier than troublesome works as shown in Fig.11.

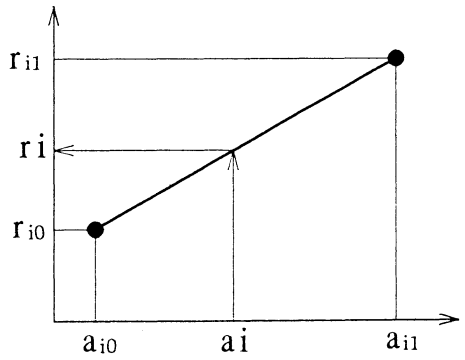


Fig.10 Relationship between analog input ai and reference ri (i=1~20)

4 Motion control of a humanoid robot

Some experiments controlling the humanoid robot by master/slave control mode using the motion capture device developed in this study were carried out. The humanoid robot is able to stand up and sit down,turn the direction of walking with two legs. Although we are taking aim at realizing walking motion of the humanoid robot with our motion capture device, but it was found to be very difficult using only position information from joints. We are considering that some other additional informations such as those from force sensor, vision sensor and so on are necessary to be introduced.

5 Conclusions

The motion capture device in this study made it possible to easily generate a sequence of references to control a humanoid robot which has very large number of DOFs. But we could not realize walking motion of humanoid robot with the motion capture device so far. We are now planning to introduce some force sensors and balance sensors into our humanoid robot in order to realize walking motion.

Acknowledgements

We thank Professor Hirochika Inoue and Associate Professor Masayuki Inaba, Faculty of Engineering, Tokyo University for their invaluable advice.

References

[1] Masayuki Inaba,"Remote Brained Robotics: Interfacing AI with Real World Behaviors",*Proceedings of the 6th International Symposium on Robotics Research(ISRR6)*,pp.335-344,(1993).

[2] Masayuki Inaba,"A study on humanoids taking remote brain approach",*Proceedings of the 12th meeting of the Robotics Society of Japan*,pp.305-306,(1994).

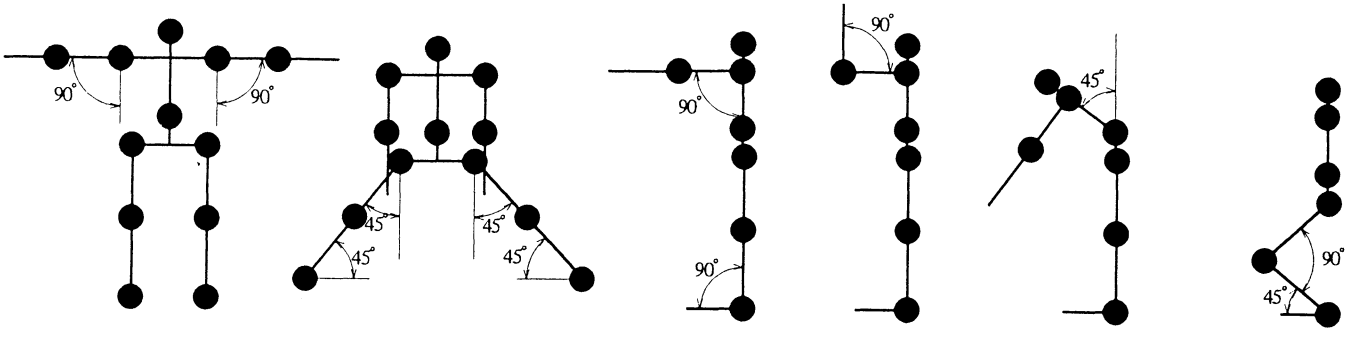


Fig.11 Basic posture for calibration of the motion capture device

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

ROBOT PATH SEARCH INCLUDING OBSTACLES BY GA

Hidehiko YAMAMOTO

Dept. of Opto-Mechatronics, Faculty of Systems Engineering, WAKAYAMA University
930, Sakaedani, Wakayama-shi, 640-8510

Abstract; *This paper describes a collision avoidance method in automatically generating a robot moving path that needs some visiting points with Lisp language. The avoidance method adopts a warning area divided a robot's surroundings into several sections and classifies the next robot action into two actions, safe moving action and collision action. The collision avoidance method is applied for a moving robot in FA factory including four machine tools and three obstacles.*

Keywords; Robot Path Planning, Genetic Algorithm, FA System, Moving Robot

1. INTRODUCTION

As one of the next generation's FA factory figures, a decentralized autonomous production system with a production line that is changeable at any time is considered. Under these circumstances, it is difficult for a pre-programmed robot to move around in a FA factory. The function to automatically generate its own path according as circumstances change is needed. Research for this function does not consider an obstacle in a robot moving space. A real FA factory includes some obstacles such as shelves and an inspection room.

This paper describes the automatic creation of a robot path which avoids an obstacle on the premise that there's a space where a robot freely moves around and there are obstacles in the space. As an application example, an automatic robot path creation in a FA factory that includes four machine tools and three obstacles is used.

2. PROBLEM ESTABLISHMENT

For the robot path search that this paper describes, we suppose a robot leaves a starting point, visits some other points and returns to its starting point in a factory space ; our goal is automatically to create such a robot moving path. This path corresponds to robot operations such that a robot arrives at the bay of a machine tool to exchange parts or tools and, after the operation, moves on to another machine tool.

So far, research can automatically create a robot moving path whereby a robot freely moves around in a moving space. However, there are usually somethings lying in a robot's moving space in a real FA factory. For example, the factory is interspersed with tool shelves and a simple inspection room. Thus, the present research assumes a free space including obstacles that is closer to a real FA factory as an object space where a robot moves around and deals with the automatic creation of the robot's moving paths in such a space. That is, the research develops an automatic method with Lisp language for creating a robot's moving paths that a robot returns to its starting point after visiting some points and negotiating some obstacles in the operation space on a FA factory.

The main theme of the research is to search a robot moving path with Lisp language not interfering with obstacles. The typical researches for a robot obstacles' avoiding are Configuration space method[1] and Potential space method[2]. Configuration space method considers a new space expressing an obstacle figure as an expanded one according as a moving object's figure and, in this new space, searches a moving path. Potential space method artificially defines a potential and searches a moving path by using the potential value and a gradient decent method. However, the obstacle's avoiding of this research is to confirm whether the next step is a step which will lead to collisions with an obstacle or not, to judge whether the step must be modified or not and then to find a moving path. In this meaning, the research is different from conventional robot obstacle avoidance methods.

3. OBSTACLE AVOIDANCE

3.1 Robot Path Expression with Lisp

This research expresses a robot moving path with nineteen Lisp function[3]. For example, **mf** is a Lisp function indicating "move forward" and **mb** is "move backward". The Lisp function **tl** means "turn left" and **tr**

means "turn right". The turning angle is defined as 30 degrees. Because of the angle definition, a robot moving directions are decided as $\mathbf{d}_\beta$ kinds (β =0, 1, ···, 11) as shown in *Fig 1*. **S01~S02** are also Lisp functions to measure the shortest distance between a robot and the obstacle which is lying ahead in the section equally divided a robot's surrounding into twelve sections.

In this way, by randomly selecting the sequence of the nineteen Lisp functions, we create a Lisp function corresponding to an individual and GA operations such as crossover, mutation and reproduction are repeated till a cycle path for a robot is completed.

3.2 Unsafe Area and Movement Modification

A robot moving path expressed in Lisp language is improved by GA operations[4]. If this improvement is executed by conventional GA operations that means creating initial populations, where crossover, mutation and reproduction are blindly repeated, the possibility occurs that many lethal genes are created. For example, consider *Fig 2* where a robot's current location is $\mathbf{P}_1$. In this case, if a Lisp program generated as the robot next moving action is (··· **(tl)(tl)(tl)(mf)**···), the robot takes the actions "turn left, turn left, turn left and move forward". Judging from the figure, the actions will make the robot collide with an obstacle.

In order to solve this problem, this research adopts new method to modify on purpose of a Lisp program generated by GA. However, when the generated Lisp program is given a partial modification, the problem then becomes which parts of the program to modify and how to modify in order not to have a collision. To solve this, a Safe moving action's program (**SMAP**) is proposed. SMAP defines an area around an obstacle as an unsafe area and modifies a Lisp function indicating a robot moving in the unsafe area. By including SMAP in GA operations, collision avoidance of a robot can be realized.

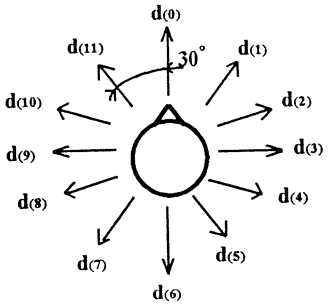


Fig 1 **Directions of robot**

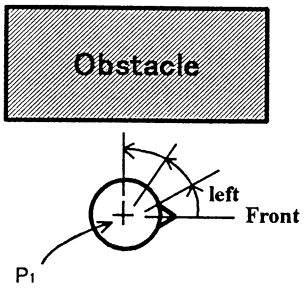


Fig 2 **Collision example**

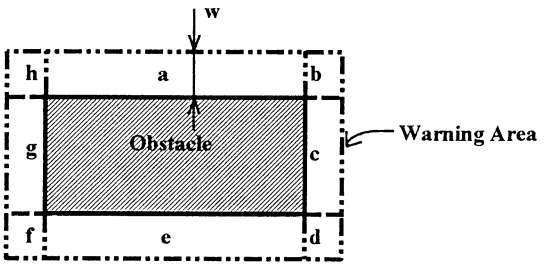


Fig 3 **Warning area example**

An unsafe area for SMAP is a section whose depth is *w* around an obstacle, as shown in *Fig 3*. The unsafe area is divided into sections along an obstacle's side. The number of sections for an obstacle that has *z* sides is considered as 2*z*. As *Fig 3* example is an obstacle with four sides, eight sections (a, b, c, d, e, f, g, h) are considered. The unsafe area gives a robot moving into the area a limitation on its next movement. It means the area plays the role of a pointer to modify on purpose a Lisp function in the Lisp program generated by GA. The following is the concrete algorithm of SMAP to modify a Lisp program by using a unsafe area. Before the algorithm is shown, some terms used in the algorithm are defined.

[Definition1] Divided unsafe area α; each unsafe area is divided into 2 z, with being the number of the obstacle's side. □

[Definition2] Movement $\mathbf{M}_{next}$; A robot's next movement, either forward (**mf**) or backward (**mb**) in direction $d_{(\beta)}$. □

[Definition3] Safe movement $\mathbf{M}_{safe}$; $\mathbf{M}_{safe}$ will not result in a collision. □

[Definition4] Safe movement set $\mathbf{S}_{safe}$; the set of $\mathbf{M}_{next}$ which will not result in a collision. is considered as $d_{(\beta)}$ kinds of moving directions. □

[**Definition5**] Divided safe movement set $S_{safe}(\alpha)$; the safe movement set S_{safe} for a robot in a given unsafe area α. □

Although a divided safe movement set $S_{safe}(\alpha)$ has different elements for each part of the unsafe area α, all of the elements are the directions where there is collision with an obstacle by carrying out M_{next}. For example, if α is e in ***Fig 3***, the moving directions $d_{(3)}$~$d_{(9)}$ correspond to $S_{safe}(e)$ because they are the directions that a robot doesn't result in collision with an obstacle.

[**Definition6**] Collision movement M_c ; M_{next} corresponds to actions where a robot will collide with an obstacle by carrying out M_{next}. □

[**Definition7**] Collision movement set S_c; the set of M_{next} which will result in a collision. □

[**Definition8**] Divided collision movement set $S_c(\alpha)$; the collision movement set S_c for a robot in a given unsafe area α. □

Although a divided collision moving actions set $S_c(\alpha)$ has different elements for each division of unsafe area α, all of the elements are the directions which will result in a collision with an obstacle by executing the action M_{next}. For example, if α is e in ***Fig 3***, the moving actions $d_{(10)}$~$d_{(2)}$ correspond to $S_c(e)$ because they are the directions in which a robot will collide with an obstacle.

[SMAP]

STEP 1 : Find a robot current location's co-ordinates $P(x_p, y_p)$.

STEP 2 : Verify the co-ordinates P and α, find α satisfying the condition

$P \in \alpha$ and perform the following rules.

IF ; α is not found, THEN ; execute the next movement M_{next} and go to STEP3.

IF ; α is found, THEN ; perform STEP 4.

STEP 3 : If a robot arrives at the goal, this algorithm comes to an end, if not,

return to STEP 1.

STEP 4 : Perform the following rules.

IF ; $M_{next} \in S_{safe}(\alpha)$, THEN ; perform M_{next} and return to STEP 3.

IF ; $M_{next} \in S_c(\alpha)$, THEN ; perform STEP 5.

STEP 5 : Randomly select an element $d_{(\beta)}$ from among a set $S_{safe}(\alpha)$ and execute

$M_{next} \leftarrow d_{(\beta)}$.

STEP 6 : Execute M_{next} and return to STEP 3. □

A Lisp program generated by random operations of GA includes possibility that the program will change into a bad one and a robot collides with an obstacle with the next step even if it has taken a good path up to now. Suddenly what had been a good individual becomes a useless one. In this case, all the program time is judged useless and thus the individual is totally scrapped. However our developed algorithm forcibly modifies a Lisp function, one part of the Lisp program generated by GA operations in STEP 4. Because of this forced modification process, the change from a suddenly known lethal gene to a good gene is performed. Because of this operation, our GA individual does not become useless and continuous use of the robot movement program is possible.

4. SIMULATION EXAMPLES

SMAP to avoid obstacles was applied to the path of a robot moving in FA factory. From the starting point, a robot visits four entrances in front of four machine tools, avoids three obstacles, two shelves and an inspection room and returns to a starting point. The example's aim is to create a path for a robot's movement. Each operator value used in GA operations of the example is individual number n=400, crossover probability=88%, mutation probability=2% and elite preservation number = 5. Also, by adopting a series of random numbers, some simulations were executed. ***Fig 4*** shows one of the simulation results. This Figure shows fitness curves from generation 0 to generation 600. A ■ mark in the figure indicates an average fitness among a population and a □ mark indicates the best fitness among a population. The figure says the higher the generation number becomes, the lower the fitness becomes.

Fig 5 shows the robot path acquired in this case. Although the paths acquired with other random number series were not quite same as the result of ***Fig 5***, these paths with other random number series could automatically find paths to avoid obstacles. The path acquired by the research is one to avoid obstacles and get close to a goal by repeating forward and backward. Though the path still includes some useless movements, it steadily avoids obstacles. It is considered that a developed method will find a better path if the problem of repeated forward and backward movement is improved.

5. CONCLUSIONS

This paper described a collision avoidance method in automatically generating a robot moving path that needs some visiting points with Lisp language. The avoidance method adopts dividing the robot's

surroundings into safe and unsafe areas and classifying the robot's next movement as safe or unsafe. It avoids obstacles by changing a robot's next movement when it enters unsafe area by modifying the original GA to a safe one.

The collision avoidance method was applied to a moving robot in a FA factory including visiting four machine tools and avoiding three obstacles. As a result, the path to visit four points by avoiding obstacles was automatically created. Though the acquired path is not a smooth one, the problem is an important future research theme.

References

1. Lozano-Perez,T(1981), Automatic Planning of Manipulator Transfer Movement, IEEE Trans. Syst. Man. Cybern., Vol. SMC-11, No.10, 681-698.
2. Barraquand,J and Latombe, JC (1990), A Monte-Carlo Algorithm for Path Planning with Many Degrees of Freedom, IEEE Int. Conf. Robotics and Automation, 1712-1717.
3. Yamamoto, H(1996), Robot Path Planning through Some Places by Genetic Algorithm, Proc. of Intr. Symp. On Artificial Life and Robotics, 47-50.
4. Goldgerg, DE(1989), Genetic Algorithm in Search, Optimization and Machine Learning, Addeson Wesely.

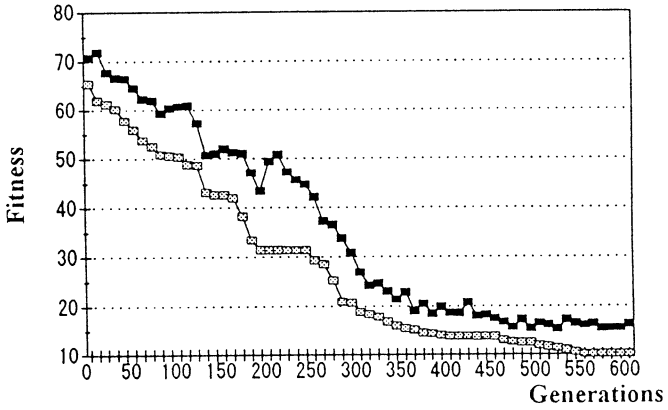


Fig 4 Fitness curves

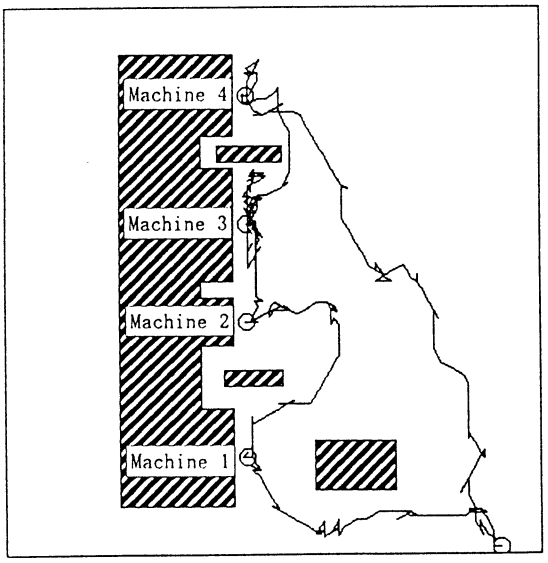


Fig 5 Simulation result

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Robot Control/Learning Scheme with Task Compatibility

Qiao Guo
Robotics research Center, Beijing Institute of Technology
Beijing, 100081, CHINA

Abstract

This paper proposes a robust decentralized control/learning scheme with task compatibility for redundant robot manipulators in task space. It can finish on-line trajectory planning and control/learning through optimizing the new type of performance index with the constrains of the control signals and the states of the system. The scheme can reduce the on-line computation quantity effectively because of using the decentralized control way. And it has both the adaptability to the changes of the task and unknown/changeable parameters of the system. This scheme may be used in the rehabilitation field to help the disabled people to improve their life quantity.

Key Words: *Task Compatibility; Rehabilitation Robotics; Robot Control*

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Fine Motion Strategy Using Skill-Based Backprojection in Consideration of Uncertainty in Control and Sensing

Akira NAKAMURA*
Takashi SUEHIRO*
*Electrotechnical Laboratory
1-1-4 Umezono, Tsukuba, Ibaraki, 305-8568 Japan

Tsukasa OGASAWARA**
Hideo TSUKUNE*
**Nara Institute of Science and Technology
8916-5 Takayamacho, Ikoma, Nara, 630-0101 Japan

Abstract

A manipulator task generally consists of a sequence of several motion primitives called "skills." Skill-based motion planning is an effective way to simulate various human tasks. When planning an assembly process, a fine motion planning technique such as the backprojection method in the configuration space is often used. At AROB 1st, we proposed fine motion planning using skill-based backprojection to handle control errors. In this paper, we represent a fine motion planning method to handle control and visual errors.

Key words: manipulation skill, fine motion planning, backprojection, skill library

1 Introduction

For robots to play a meaningful part in several fields, it is necessary for them to perform various tasks using special manipulator techniques. By analyzing human task motions such as assembly and disassembly, it can be seen that movements consist of several significant motion primitives. We called these "skills" and have shown that most of the tasks of a manipulator can be composed of sequences of skills[1,2]. In other words, we demonstrated that the concept of skill is very useful for robots to achieve various human tasks.

Fine motion planning in a configuration space has been studied as a method of artificial intelligence. Various techniques of fine motion planning are often used to plan tasks such as assembly and disassembly processes. In a configuration space the planning is simplified, since an object is represented as a point. Lozano-Perez et al. proposed the concept of pre-image, and performed planning using generalized damping while accounting for sensor and control uncertainty[3]. Erdmann proposed the backprojection method, in which the goal region is projected in reverse using an error cone, which made it easier to obtain the reachable region to the goal than using the pre-image[4].

At the First International Symposium on Artificial Life and Robotics (AROB 1st), we proposed a fine motion planning method based on backprojection for various tasks composed of several manipulation skills[5]. We showed that general and skillful planning can be derived with ease. However, we dealt with control errors of the manipulator only as an uncertainty, though sensor and modeling errors may impact the reliability of task achievement.

In this paper, we propose a fine motion planning method using skill-based backprojection to effectively handle not only control errors but also visual sensor errors. In this method, we assume that hand-eye is used, since it is necessary to obtain range data from the most suitable position to assure high reliability of the task achievement.

In the next section, we explain the hierarchical structure of our manipulation system: task level, skill level, and servo level. Manipulation skills are explained in section 3. The procedures of visual sensing and geometric modeling are explained for each level in section 4. The fine motion strategy method is demonstrated by using an example of a peg-in-hole task in section 5.

2 Construction of Manipulation System

To overcome errors both in the environment model and in the manipulator motion, we developed a new concept of manipulation skill[1,2]. A skill is implemented as a sequence of motion primitives in a hybrid position-velocity-force control scheme. The manipulation system is composed of three levels: task, skill and servo layers, as follows (Fig.1).

(1) Task Level

This level is the highest layer at which a specific task such as peg-in-hole, pick-and-place, and fastening or unfastening of bolts is performed.

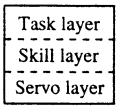
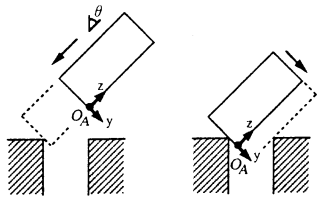


Fig.1 Hierarchy of Manipulation System

(2) Skill Level

To execute a manipulation task, the motions can be broken down into several motion primitives, each of which has a particular target state. We call these primitives "skills", and in each motion primitive there is a transition from one state to another state[1,2]. Most manipulation



(a) Move-to-touch$_f$ skill (b) Move-to-touch$_c$ skill

Fig.2 Move-to-touch skills

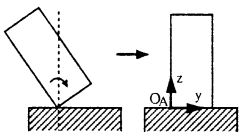


Fig.3 Rotate-to-level skill

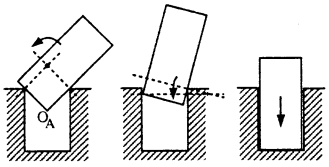


Fig.4 Rotate-to-insert skill

tasks can be achieved by a combination of skills. The skill layer is between the servo layer and the task program layer.

(3) Servo Level

This level is the lowest layer and it is where servo parameters are directly controlled.

By positioning the skill layer between the task and servo layer, it becomes possible to program a task as a sequence of skills, independent of the control of the servo layer.

3 Manipulation Skills

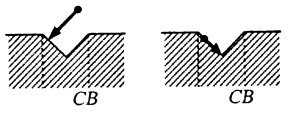
This section will explain our concept of skills. See References 1,2 for more detail.

3.1 Skill primitives

In assembly and disassembly tasks, skills in which the contact states vary are particularly significant. In this paper, we consider three skills which play an important part in such tasks: move-to-touch, rotate-to-level and rotate-to-insert. We view skill motions as occurring in two-dimensional environments.

(1) Move-to-touch Skill

The move-to-touch skill is the transition from free to vertex-to-face contact between a grasped object and another object in velocity control mode (Fig.2(a)). The similar transition of keeping contact in a different direction of motion is also part of this skill (Fig.2(b)). These are represented by the move-to-touch$_f$ skill and the move-to-touch$_c$ skill, respectively.



(a) Move-to-touch$_f$ skill (b) Move-to-touch$_c$ skill

Fig.5 Move-to-touch skills in C-space

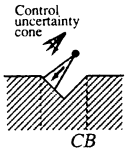


Fig.6 Move-to-touch$_f$ skill with control uncertainty

(2) Rotate-to-level Skill

The rotate-to-level skill is the transition from vertex-to-face contact to edge-to-face contact (Fig.3).

(3) Rotate-to-insert Skill

In an insertion task, it is generally difficult to achieve the state of Fig.4(b) directly when the clearance is small. The state of Fig.4(a) is achieved first by using other skills such as the skill sequence of Fig.2(a) and (b). The state of Fig.4(b) is then accomplished by gradually raising the object while maintaining the contact state of Fig.4(a). The rotate-to-insert skill is this motion of rotating the object obliquely into the hole to accurately insert it. In our study, we assume that the rotate-to-insert skill also includes the pressing motion required to achieve the goal of the insertion task (Fig.4(c)).

3.2 Skills in Configuration Space and Backprojection

Next, we will discuss the trajectory of skill motions in configuration space.

(1) Move-to-touch Skill

In the configuration space, the trajectories of the object being manipulated with the move-to-touch$_f$ skill in Fig.2(a) and the move-to-touch$_c$ skill in Fig.2(b) are drawn in Fig.5(a), (b), respectively. To take into account the uncertainty of control when using the move-to-touch$_f$ skill, we have drawn trajectory using the control uncertainty cone in Fig.6.

(2) Rotate-to-level Skill

Assuming that reference point O_A is a vertex in prior contact with the surface (Fig.3), the position of O_A on the YZ-plane in the configuration space stays constant (Fig.7).

(3) Rotate-to-insert Skill

The trajectory of the object in the configuration space when manipulated with the rotate-to-insert skill is shown in Fig.8. The transfer motion of the vertex from Fig.4(a) to Fig.4(b) is done at orientation $\theta = \theta_t$ in Fig.8, where the phase of C-obstacle CB changes.

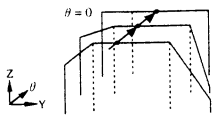


Fig.7 Rotate-to-level skill in C-space

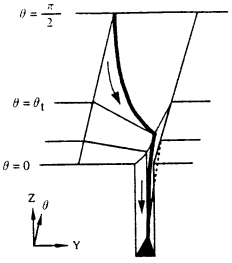


Fig.8 Rotate-to-insert skill in C-space

The backprojection of each skill is derived by the reverse trajectory from each goal[4].

3.3 Construction of Skill Library and Composition of Task

First, a skill library which consists of skill primitives expressed by trajectories in the configuration space is constructed in advance.

Next, the skill sequence is derived for a specific task. Some skill which can be performed with configuration obstacles is selected, and a backprojection is drawn by projecting in reverse the trajectory from the goal. Then, using the derived region as a subgoal, another backprojection is drawn. By repeating this procedure, a skill sequence composing the task and initial position and orientation from which the object can reach the goal can be obtained.

4 Visual Sensing and Modeling

We will now explain the scheme of measurement and geometric modeling for planning in the task and skill levels and for control in servo level. At AROB 1st, we demonstrated that manipulation planning that takes into account the uncertainty of control for a geometrically modeled working environment could be precisely achieved. If range data is obtained correctly and the environment model is constructed exactly, overall manipulation planning can be carried out only by initial visual sensing. In practice, however, visual sensing errors cannot be ignored. Even if a geometric model which exactly expresses the shape of each object is obtained, the environment model will include uncertainty because of visual sensing errors, so the reliability of the manipulation degrades. In general, range data obtained by vision systems such as a stereo camera or range finder will have large errors in depth data. Therefore, our planning of manipulation and visual sensing take into account such visual errors to maintain higher manipulation reliability.

In this study, we perform measurement and modeling for each module just before task planning, skill planning or servo control, as follows (Fig.9).

4.1 Measurement and Modeling in Task Level

In the task level, the sequence of skill primitives composing the task is decided and backprojection is derived. Measurement and modeling in this level are carried out to decide global arrangement of the objects in the working environment of the robot. Since global and rough data is used, the environment model often has some uncertainty. Thus, the backprojection derived in this level likewise may have some uncertainty.

4.2 Measurement and Modeling in Skill Level

In the skill level, the initial position and orientation of a grasped object is derived for each skill primitive using the backprojection method. Then, the backprojection is revised using a vision system to obtain local and precise data.

In this study, we assume that a removable vision system such as hand-eye can be used. Range data is measured from the position where the task-related surface[6] can be observed, and the geometric model is constructed. Since the range data of surfaces directly related to manipulation is used for modeling, reliable skill planning can be achieved. Furthermore, visual sensing is performed in each skill primitive in the direction as depth errors do not affect the attainability of skill motion.

4.3 Measurement and Modeling in Servo Level

It is desirable to make sure that the grasped object exists in the backprojection region before performing servo control. Visual sensing is carried out from the position where the surfaces of objects related to the manipulation can be observed and depth errors will have no impact on the attainability.

5 Motion Strategy and Example

We will now explain our motion strategy using an example of a peg-in-hole task (Fig.10).

(Step 1) Task planning

First, visual sensing for the working environment of robot is performed from some viewpoint. Modeling is performed for visible or obscured task-related surfaces[6]. If the hole of object Q can be seen, model matching can be carried out exactly by using data around the hole (Fig.11(a)). If the hole cannot be seen, model matching can be carried out by using the outer surfaces (Fig.11(b)), but the reliability of the model will be inferior around the

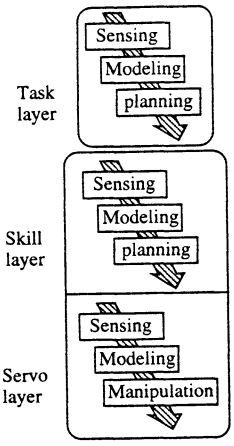


Fig.9 Flow in each level

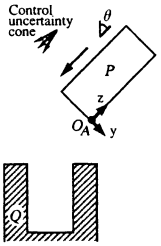


Fig.10 Peg-in-hole task

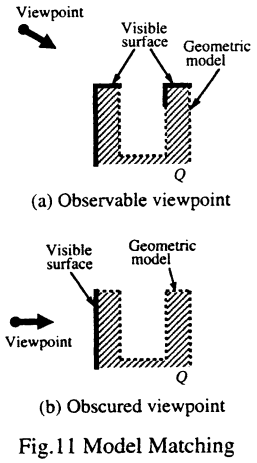


Fig.11 Model Matching

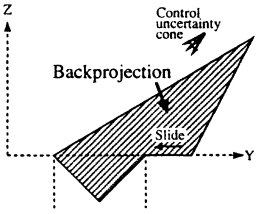


Fig.12 Backprojection

hole. Next, the sequence of skill primitives is derived. There are various methods with which the sequence could be constructed. We assume that the following sequence is specified by the operator.

(i) $Skill_1$: move-to-touch skill in -z-direction (including transference to initial state by position control)
(ii) $Skill_2$: move-to-touch skill in y-direction
(iii) $Skill_3$: rotate-to-insert skill

Backprojection is derived using the reverse sequence of these three skill primitives (Fig.12). If the hole cannot be seen, the reliability of the backprojection will be inferior.

(Step 2.1) Skill planning of $Skill_1$

The data of the task-related surface around the hole is obtained precisely by using hand-eye. Visual sensing is carried out as close as possible from the direction of motion of object P to avoid any depth errors in the visual data. Next, the model of object Q is reconstructed, and the backprojection is revised.

(Step 2.2) Servo control of $Skill_1$

After moving to the initial state, range data is obtained to make sure that object P actually exists in the backprojection. Visual sensing is carried out from the viewpoint where both objects are observable and the depth error has little influence, and then modeling is performed. If object P is not located in the backprojection, its position is revised. Then, servo control is performed.

(Step 3) Skill planning and servo control of $Skill_2$

The model of object Q is constructed by local visual sensing and the backprojection is derived. Next, the model of both objects is constructed by sensing from the position where both can be seen to make sure that the state is adequate. Then, manipulation is performed.

(Step 4) Skill planning and servo control of $Skill_3$

The same procedure described for $Skill_2$ in Step 3 is repeated.

6 Conclusion

We have described fine motion planning using skill-based backprojection that takes into account the uncertainty in control and visual sensing. In our method, just before each step in task planning, skill planning and servo control, visual sensing and geometric modeling are performed. These are carried out taking account of each situation and purpose. Therefore, highly reliable task achievement can be accomplished with our method. In the future, we will study the derivation of more appropriate position and orientation of the visual sensing.

Reference

1. Suehiro T, Takase K (1990) Skill based manipulation system (in Japanese). J Robotics Soc Jpn 8:551-562
2. Hasegawa T, Suehiro T, Takase K (1991) A model-based manipulation system with skill-based execution in unstructured environment. In: Dario P (ed) Proceedings of the 5th International Conference on Advanced Robotics, Pisa, Italy pp 970-975
3. Lozano-Perez T, Mason MT, Taylor RH (1984) Automatic synthesis of fine-motion strategies for robots. Int. J. Robotics Res 3:3-24
4. Erdmann M (1986) Using backprojections for fine motion planning with uncertainty. Int. J. Robotics Res 5:19-45
5. Nakamura A, Ogasawara T, Suehiro T, Tsukune H (1996) Fine motion strategy for skill-based manipulation. In: Sugisaka M (ed) Proceedings of the International Symposium on Artificial Life and Robotics (AROB 1), Beppu, Oita, Japan, Feb 18-20, 1996, pp 158-161
6. Nakamura A, Ogasawara T, Tsukune H, Oshima M (1996) Surface-Based Geometric Modeling Using Task-Oriented Teaching Trees. In: Asada M (ed) Proceedings of the International Conference on Intelligent Robots and Systems (IROS 96), Osaka, Japan, Nov 4-8, 1996, pp 1015-1022

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Sliding Mode Controller for Robot Manipulators with Predetermined Transient Response

Kang-Bark Park, Teruo Tsuji
Dept. of Electrical Engineering
Kyushu Institute of Technology
1–1 Sensui-cho, Tobata-ku
Kitakyushu 804–8550 JAPAN

Ju-Jang Lee
Dept. of Electrical Engineering
Korea Advanced Institute of Sci. and Tech.
373–1 Kusong-dong, Yusong-gu
Taejon 305–701 KOREA

Abstract

In this paper, a sliding mode control scheme is proposed for robot manipulators. By using a novel sliding hyperplane, the reaching phase problem is overcome, i.e., the closed-loop system always shows the invariance property to parameter uncertainties and external disturbances. Furthermore, a predetermined transient response can be cbtained.

1 Introduction

Sliding mode control, which is one of the robust control methods, has been studied a lot due to the advantage that it can be applied to the system with nonlinearity, uncertainties, and bounded external disturbances [1]. It is designed for the system state to be forced to stay on the predetermined sliding surface. When the system is in the sliding mode, the output response shows the invariance property to parameter uncertainties and external disturbances. In the actual case, however, a system state may not be on the sliding surface at an initial instant. It makes a "reaching phase." During the reaching phase, the output response is sensitive to parameter variations and external disturbances. In addition, in this period, the output trajectory cannot be predetermined.

In order to overcome the reaching phase problem, a rotating sliding hyperplane [2] and a moving sliding surface [3] were proposed. Those methods, however, have some difficulties to apply when the initial state is in the I or III quadrant and for the moving scheme of discrete way.

In order to overcome these problems, a novel sliding mode control scheme is proposed in this paper. By using a novel sliding hyperplane, the reaching phase is totally eliminated wherever an initial state is located. Thus, the overall system is always in the sliding mode and shows the invariance property to parameter variations and external disturbances all the time. It is also guaranteed that the output tracking error can be totally determined in advance.

2 Problem Formulation

The dynamic equation for an n-link robot manipulator is described by

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = u + d, \tag{1}$$

where $M(q)$ is an $n \times n$ inertia matrix, $C(q,\dot{q})$ is an $n \times n$ matrix corresponding to Coriolis and centrifugal factors, $g(q)$ is an $n \times 1$ vector of gravitational forces, d is an $n \times 1$ bounded disturbance vector, q is an $n \times 1$ joint angular position vector, and u is an $n \times 1$ input torque vector. It should be noted that the matrix $\dot{M}(q) - 2C(q,\dot{q})$ is skew-symmetric with a suitable definition of $C(q,\dot{q})$ [4].

Let us define each matrices as $M(q) = M^{\circ}(q) + \Delta M(q)$, $C(q,\dot{q}) = C^{\circ}(q,\dot{q}) + \Delta C(q,\dot{q})$, $g(q) = g^{\circ}(q) + \Delta g(q)$, where "$\circ$" denotes the nominal value and "Δ" is the estimation error. It is assumed that $\Delta M_{ij}, \Delta C_{ij}$, and Δg_i are bounded by M_{ij}^m, C_{ij}^m, and g_i^m respectively as $|\Delta M_{ij}(q)| \le M_{ij}^m(q)$, $|\Delta C_{ij}(q,\dot{q})| \le C_{ij}^m(q,\dot{q})$, $|\Delta g_i(q)| \le g_i^m(q)$, where "$m$" represents the maximal absolute estimation error of each element, and $i, j = 1, 2, \cdots, n$. It is also assumed that $|d_i| \le d_i^m$, where $i = 1, 2, \cdots, n$.

3 Design of Control System

Let us define a novel sliding hyperplane as

$$s(t) = \dot{e}(t) + \Lambda e(t) - h(t), \tag{2}$$

where $e(t) = q(t) - q_d(t)$, $q_d(t)$ is a given twice continuously differentiable reference trajectory, $s \in \Re^n$, $\Lambda =$

$diag(\lambda_1, \lambda_2, \cdots, \lambda_n) \in \Re^{n\times n}$, $\lambda_i > 0$, and $h(t) \in \Re^n$ satisfies the following assumption.

Assumption 1 *$h_i : \Re_+ \rightarrow \Re$, $h_i \in C^1[0,\infty)$, $\dot{h}_i \in L^\infty$, and $h_i(0) = \dot{e}_i(0) + \lambda_i e_i(0)$, where $C^1[0,\infty)$ represents the set of all first differentiable continuous functions defined on $[0,\infty)$ and $i = 1, 2, \cdots, n$.*

Since the augmented function $h(t)$ can be designed arbitrarily, it is always possible to design $h(t)$ in order that the Assumption 1 holds.

Remark 1 *From the Assumption 1 and the definition of the proposed sliding hyperplane (2), the system state is on the sliding hyperplane at an initial instant, i.e.,*

$$s(0) = 0. \tag{3}$$

The assumption that an initial condition is available is not restrictive because the measured and/or estimated data of the system state can be obtained at each sampling time. Especially, for robot manipulators, one can get the position and the velocity data at each sampling time. Thus, the initial condition can be regarded as the data obtained when the control system starts to operate. In fact, a lot of previous works proposed to overcome the reaching phase problem have used an initial condition of the system state [2]-[3].

Based on the Remark 1 , the following theorem can be derived for the sliding mode occurrence.

Theorem 1 *If the following control system (4)*

$$u = u_{eq} - Sgn(s)\cdot K, \tag{4}$$

is applied to the robot manipulator (1), the closed-loop system is in the sliding mode all the time, i.e.,

$$s(t) \equiv 0 \qquad \forall\, t \geq 0, \tag{5}$$

where u_{eq} represents the equivalent control, $u_{eq} = M^0\left(\ddot{q}_d + \dot{h} - \Lambda\dot{e}\right) - C^0\,(s - \dot{q}) + g^0$, $K = M^m|\Lambda\dot{e} - \dot{h} - \ddot{q}_d| + C^m|s - \dot{q}| + g^m + d^m + \eta$, $\eta = [\ \eta_1,\ \eta_2,\ \cdots,\ \eta_n\ ]^T$, $\eta_i > 0$, $Sgn(s) = diag\{\ sgn(s_1),\ sgn(s_2),\ \cdots,\ sgn(s_n)\ \}$, and the absolute of a vector denotes the vector whose element has its absolute value, i.e., $|x| = [\,|x_1|,\ |x_2|, \cdots, |x_n|\,]^T$.

Proof Let us define the following positive definite function as a Lyapunov function candidate:

$$V = \frac{1}{2}s^T M s. \tag{6}$$

Differentiating (6) with respect to time, adopting the skew-symmetricity of $\dot{M}(q) - 2C(q,\dot{q})$, and substituting the control law (4) into (??) one can obtain

$$\begin{aligned}
\dot{V} &= s^T M\dot{s} + s^T C s = s^T\left(M\dot{s} + Cs\right) \\
&= s^T\left(u + d - g + M\left(\Lambda\dot{e} - \dot{h} - \ddot{q}_d\right) + C\,(s - \dot{q})\right) \\
&= s^T\Big\{\left(M - M^0\right)\left(\Lambda\dot{e} - \dot{h} - \ddot{q}_d\right) + \left(C - C^0\right)(s - \dot{q}) \\
&\quad - Sgn(s)\cdot M^m\left|\Lambda\dot{e} - \dot{h} - \ddot{q}_d\right| \\
&\quad - Sgn(s)\cdot C^m\,|s - \dot{q}| + \left(g^0 - g\right) - Sgn(s)\cdot g^m \\
&\quad + d - Sgn(s)\cdot d^m - Sgn(s)\cdot\eta\Big\} \\
&\leq -\sum_{i=1}^{n}\eta_i\,|s_i|\,.
\end{aligned} \tag{7}$$

From (6) and (7), it is clear that V is a positive definite function and $\dot{V}$ is a negative definite function, i.e., V is a Lyapunov function. And $V(0) = 0$ because $s(0) = 0$ from (3). Therefore, V is always zero, i.e., $V(t) \equiv 0 \quad \forall\, t \geq 0$. Since V is a positive definite function of s, it is equivalent to $s(t) \equiv 0 \quad \forall\, t \geq 0$. ■

Remark 2 *Since $s(t) \equiv 0 \quad \forall\, t \geq 0$ from (5), there is no reaching phase and the overall system is in the sliding mode at all times. It also implies that the closed-loop system always shows the invariance property to parameter uncertainties and external disturbances. Furthermore, it can be easily known that the system output is governed by the following equation all the time,*

$$\dot{e}(t) + \Lambda e(t) = h(t), \qquad \forall\, t \geq 0, \tag{8}$$

that is, the output tracking error $e(t)$ can be totally predetermined and it is not affected by parameter variations and external disturbances at all.

Since the dynamic equation of the overall system is equivalent to (8), the following theorem can be derived.

Theorem 2 *For the robot manipulator (1) applied by the control system (4), the closed-loop system is globally exponentially stable if the augmented function $h_i(t)$ is designed such that there exist positive constants α and β satisfying*

$$|h_i(t)| \leq \alpha\cdot e^{-\beta t}, \qquad \forall\, t \geq 0, \tag{9}$$

where $i = 1, 2, \cdots, n$.

Proof Since it is so obvious, the proof is omitted. ■

Now, let us define $h(t)$ as the following equation in order that the output tracking error converges to zero *in a finite time*:

$$h(t) = \dot{f}(t) + \Lambda f(t), \tag{10}$$

where the function $f(t) = [\ f_1, f_2, \cdots, f_n\ ]^T \in \Re^n$ satisfies the following assumption.

Assumption 2 $f_i : \Re_+ \rightarrow \Re$, $f_i \in C^2[0,\infty)$, $\dot{f}_i, \ddot{f}_i \in L^\infty$, *the support of* f_i *is the bounded interval* $[0, T_f]$ *for some* $T_f > 0$, $\dot{f}_i(0) = \dot{e}_i(0)$, *and* $f_i(0) = e_i(0)$, *where* $C^2[0,\infty)$ *represents the set of all second differentiable continuous functions defined on* $[0,\infty)$ *and* $i = 1, 2, \cdots, n$.

Remark 3 *It is obvious that the augmented function* $h(t)$ *defined in (10) satisfies the Assumption 1.*

Substituting (10) into (2), one can get

$$s = \dot{e} + \Lambda e - h = \dot{e} + \Lambda e - \left(\dot{f} + \Lambda f\right) = \dot{\epsilon} + \Lambda \epsilon, \tag{11}$$

where $\epsilon(t) = e(t) - f(t)$.

Remark 4 *From the Assumption 2 , it is obvious that the following equations are satisfied.*

$$\epsilon(0) = 0 \quad \textit{and} \quad \dot{\epsilon}(0) = 0. \tag{12}$$

Therefore, the following theorem can be derived.

Theorem 3 *If the control system (4) with the sliding hyperplane (11) is applied to the robot manipulator (1), the output tracking error* $e(t) = q(t) - q_d(t)$ *goes to zero in a finite time* T_f.

Proof From (5), (11), (12), and $\epsilon(t) = e(t) - f(t)$, it is obvious that

$$e(t) \equiv f(t), \qquad \forall\, t \geq 0. \tag{13}$$

Thus, from Assumption 2 the output tracking error $e(t)$ goes to zero in a finite time T_f, i.e.,

$$e(t) = 0, \qquad \forall\, t \geq T_f.$$

■

Remark 5 *From (13), it is obvious that the output tracking error trajectory can be entirely determined in advance. Thus, the design of* $f(t)$ *is identical to that of the output tracking error* $e(t)$, $\forall\, t \geq 0$.

Remark 6 *There are lots of methods to design* $f(t)$ *satisfying the Assumption 2. A cubic polynomial is one of them. For example, the following cubic polynomial can be used as* $f(t)$ *for given conditions,* T_f, $f(0) = e(0)$, $\dot{f}(0) = \dot{e}(0)$, *and* $f(T_f) = \dot{f}(T_f) = 0$.

$$f(t) = \begin{cases} a_0 + a_1 \dfrac{t}{T_f} + a_2 \left(\dfrac{t}{T_f}\right)^2 + a_3 \left(\dfrac{t}{T_f}\right)^3 & 0 \leq t \leq T_f, \\ 0 & t > T_f, \end{cases} \tag{14}$$

where $a_0 = e(0)$, $a_1 = T_f \dot{e}(0)$, $a_2 = -3e(0) - 2T_f \dot{e}(0)$, *and* $a_3 = 2e(0) + T_f \dot{e}(0)$.

4 Illustrative Example

The simulation has been carried out for a two-link robot manipulator model used by Yeung and Chen [5]. The parameter values are also the same as those of Yeung and Chen. In the simulation, $\lambda_1 = \lambda_2 = 3$, and $T_f = 1.0$ second were used, and the augmented function $f(t)$ was chosen as (14).

Figure 1 shows reference and actual trajectories for both joints. The solid line shows the actual output trajectory for joint 1, and the dotted line is that of joint 2. The dashed line represents the reference trajectory for joint 1, and the dash-dot line is that of joint 2. As can be seen in this figure, outputs for both joints track the reference trajectories.

The output tracking errors for both joints are presented in Fig. 2. One can easily know that they converge to zero in $T_f = 1.0$ second. In other words, after the relaxation time T_f that can be set arbitrarily, system outputs track reference trajectories perfectly.

Figure 3 shows the sliding hyperplane variable $s(t)$. As can be seen in this figure, system state is always on the sliding hyperplane, i.e., the overall system is in the sliding mode all the time.

The control input is shown in Figure 4. Since the system is always in the sliding mode, it shows the chattering phenomenon all the time. This phenomenon can be avoided by using the saturation function instead of the switching function.

5 Conclusions

In this paper, the sliding mode control scheme using the novel sliding hyperplane has been proposed for robot manipulators. By using the proposed method, the reaching phase problem was eliminated, and it has been shown that the closed-loop system is globally exponentially stable. It was also guaranteed that the

output tracking error converges to zero *in a finite time* that can be set arbitrarily, and the transient response can be predetermined.

Acknowledgements

This work was supported by Kyushu Electric Power Co.; Inc.

References

[1] J. Y. Hung, W. Gao, and J. C. Hung, "Variable structure control: A survey," *IEEE Trans. Ind. Electron.*, vol. IE-40, no. 1, pp. 2–22, 1993.

[2] F. Harashima, H. Hashimoto, and K. Maruyama, "Sliding mode control of manipulator with time-varying switching surfaces," *Trans. of the Society of Instrument and Control Engineers*, vol. 22, no. 3, Mar., pp. 335–342, 1986.

[3] S. B. Choi, C. C. Cheong, and D. W. Park, "Moving sliding surfaces for robust control of second-order variable structure systems," *Int. J. Contr.*, vol. 58, no. 1, pp. 229–245, 1993.

[4] D. Koditschek, "Adaptive Strategies for the Control of Natural Motion," *Proc. of the 24th Conf. on Decision and Control*, Fort Lauderdale, Fla., December, 1985.

[5] K. S. Yeung and Y. P. Chen, "A new controller design for manipulators using the theory of variable structure systems," *IEEE Trans. Automat. Contr.*, vol. AC-33, no. 2, pp. 200–206, 1988.

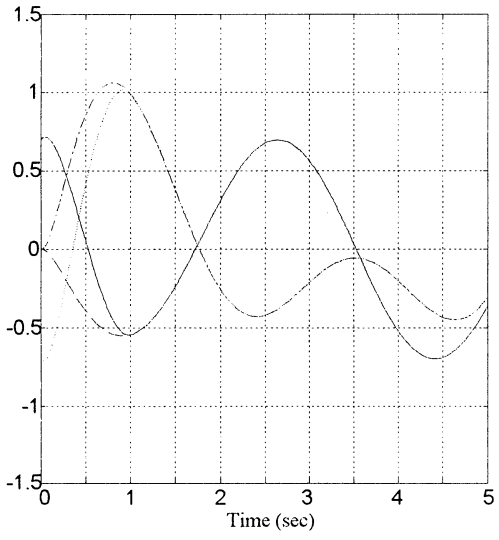


Fig. 1. Desired and Actual Trajectories

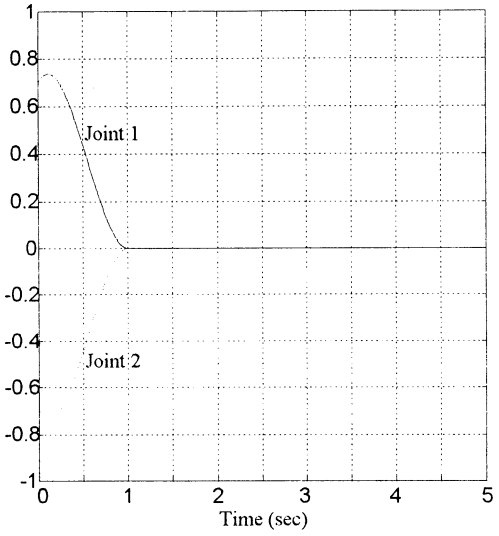


Fig. 2. Output Tracking Errors

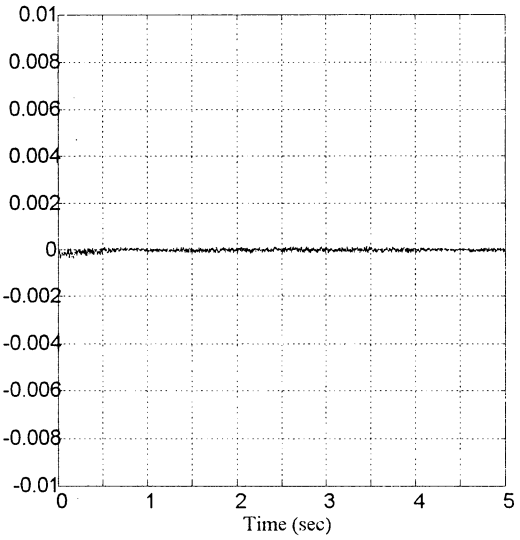


Fig. 3. Sliding Hyperplane (s)

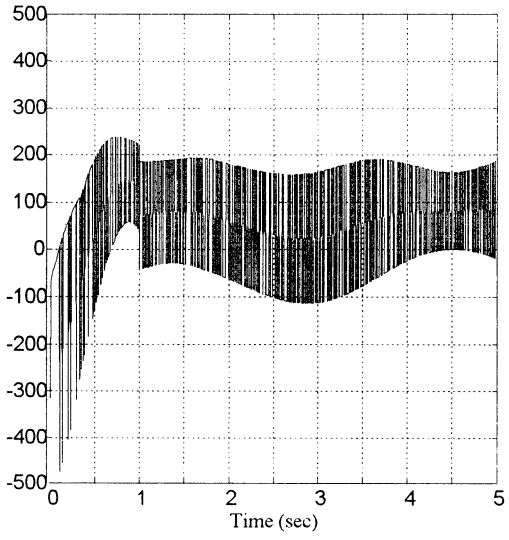


Fig. 4. Control Input Torque (N.m)

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Novel Application of Legged Mobile Robot to Human Robot Collaboration

Hiroshi Mizoguchi Yoshiyasu Goto Ken-ichi Hidai Takaomi Shigehara Taketoshi Mishima

Faculty of Engineering
Saitama University
255 Shimo Okubo, Urawa 338-8570, Japan

Abstract

This paper proposes a human following robot as a novel human collaborative robot. The human following robot is such legged mobile robot that is possible to follow the person utilizing its vision. Towards future aging society, human collaboration and human support are required as novel applications for robots. Such human collaborative robots share the same space with human. But conventional robots are isolated from humans and lack the capability to observe humans. Study on human observing function of robot is crucial to realize novel robot such as service and pet robot. To collaborate and support humans properly human collaborative robot must have capability to observe and recognize humans. The authors are currently implementing a prototype of the proposed following robot. As a base for the human observing function of the prototype robot, we have realized face tracking utilizing skin color extraction and correlation based tracking. We also develop a method for the robot to pick up human voice clearly and remotely by utilizing microphone arrays. Results of these preliminary study suggest feasibility of the proposed robot.

1 Introduction

Towards future aging society, expectation of human collaborative robot has been raised recently[1]-[4]. There are strong expectation and social demand for human collaboration and human support by robot. If a robot could interact with human, follow his/her order, recognizes its environment and support him/her, it would be very convenient and much desirable for us. This paper proposes a human following robot as a novel human collaborative robot. The human following robot is such legged mobile robot that is possible to follow the human "master" utilizing its vision. Human collaborative robot must share the same space with human. But conventional robots are isolated from humans and lack the capability to observe humans.

To collaborate and support humans properly the human collaborative robot must have capability to observe and recognize humans. Study on human observing function of robot is crucial to realize novel robot such as service and pet robot. Ultimate goal of the study is to realize such robot that can accompany by simply putting baggage on it. It is not necessary for the human user of the robot to program nor teach its motion trajectory in detail. State-of-the-art vision, locomotion, and other elemental technologies are worth matured to confirm feasibility of the study.

At the time of writing this paper, the authors are constructing a prototype of the proposed following robot. As a base for the human observing function of the prototype robot, we realize robust face tracking utilizing skin color extraction and correlation based tracking[5]. We also develop a method for the robot to pick up human voice clearly and remotely[6]. The method utilizes microphone arrays. By setting gain and delay of each microphone enables to form "acoustic focus" at person's mouth. Results of these preliminary study suggest feasibility of the proposed robot.

In this paper we describe problems and required functions to realize the human following robot and its structure to meet the requirements. We also describe design philosophy of the prototype robot and progress status of the constructing robot. In the following, section 2 describes required functions. In section 3, implementation issues of the current prototype are described. Section 4 is conclusion.

2 Required Functions

Fig. 1 illustrates functions required to realize the proposed robot. They are mobility, visual processing, auditory processing, tactile sensation, real-time behavior planning, and real-time control.

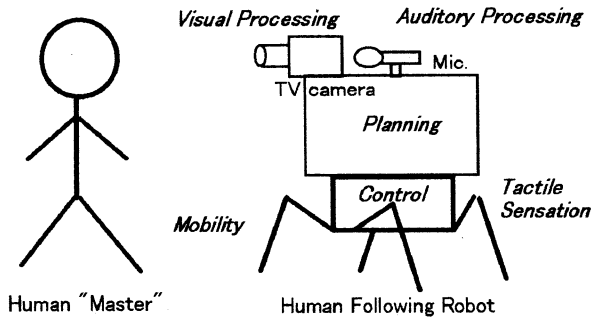


Fig. 1 Required Functions for Human Following Robot

Mobility

Mobility is quite essential for the proposed robot to follow human. Since human beings have physical bodies, live and behave in three dimensional physical world, not in logical nor virtual world. Therefore the proposed robot should also have its own body and capability to move and follow the human in the real world.

In ordinary environment where humans behave, there are obstacles, stairs and steps. Therefore wheeled locomotion mechanism has limitation for the following robot that must work around humans. In such environment, legged locomotion mechanism is more desirable for the robot to work in the humans environment.

Visual Processing

To follow a human, the proposed robot is necessary to have functions to detect and locate him visually. Once the human to be followed is detected, then tracking him should be also performed visually. Thus real-time vision and tracking function is also required.

To make visual processing robust, color image processing, such as skin-color extraction, is effective and necessary. As higher level visual functions, there are face identification and gesture recognition. They utilize results of human detection and tracking functions. Face identification enables the robot to react specified persons, such as human "master". By gesture recognition, humans can instruct or interrupt the robot's behavior interactively. These interaction can be basis for visual learning capability.

Auditory Processing

If the proposed robot could listen human voice command and behave properly, it would be very convenient and much desirable for us. But even state-of-the-art speech recognition technology requires high quality noiseless input. User of current speech recognition system must put headset microphone to prevent background noise. Because of this difficulty in usage the speech recognition still cannot be applied to the human collaborative robot.

To make the speech recognition practically applicable to the robot, some novel technique to pick up speech sound clearly and remotely is keenly required. In other words, a technique to form "acoustic focus" is needed. To realize such technique there are two problems to be solved. One is how to form the acoustic focus in three dimensional space. The other is how to track the human's mouth in real time. Location of the mouth is not fixed according to motion of human's head and body.

Tactile Sensation

Tactile sensation at sole of the robot legs is useful for the proposed robot to move around and work in the ordinary human living environment. Utilizing the tactile sensation, the robot is possible to detect steps, stairs and gaps in the environment robustly besides vision.

Besides the gesture recognition, tactile sensation of head, back and behind is also effective for human to teach and train behavior of the robot intuitively and interactively. .Human is possible to treat and train the robot as if it were a dog or an infant.

Planning and Controlling Behavior

To follow human the proposed following robot always controls orientation of its body in order that tracked image of the human is being kept at center of the visual field. Degree of rotation is based upon trajectory of the tracked target in short term memory.

3 Implementation

At the time of writing this paper, the authors are implementing a legged mobile robot with vision as a prototype of the proposed human following robot. This section describes implementation issues of the prototype in detail.

Hardware

Fig. 2 illustrates hardware block diagram of the implementing robot. It consists of several parts corresponding to required functions described in the previous section.

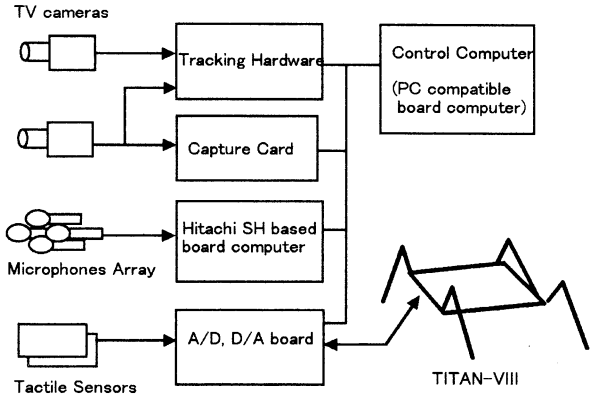


Fig. 2 Block Diagram of Implementing Prototype Robot

Mobility

To realize the mobility, we utilize TITAN-VIII[7], a research purpose four legged mobile robot kit, as a platform for the implementing robot. Fig. 3 shows outlook of the legged robot. TITAN-VIII is originally designed and developed by Prof. S. Hirose of Tokyo Institute of Technology, Japan and commercialized by a small Japanese company, Tokyo Seiki Corp.

Each leg of the robot consists of three joints and has 3 D.O.F. Therefore the robot has 12 D.O.F. in total. These twelve joints are all rotational and are actuated by DC-motor which torque can be controlled via software. Twelve potentiometers are equipped at each joint and current rotational angles of the joint can be obtained by software. Power amplifiers for the DC-motors and operational amplifiers for the potentiometers are equipped with TITAN-VIII. Thus the robot is interfaced with control computer via A/D and D/A converters.

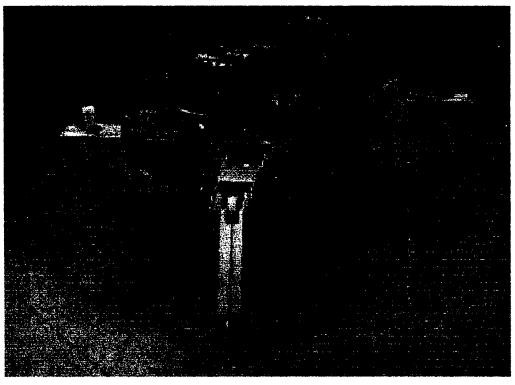


Fig. 3 Outlook of TITAN-VIII

Visual Processing

Two color CCD cameras are equiped as input device for vision. As shown in the fig. 2, output signals from the cameras are fed into both video capture card and tracking hardware. The video capture card is manufactured by Argo Craft, Tokyo, Japan. The card bundles Linux, FreeBSD and Windows NT drivers. And the Argo Craft opens source code of these drivers freely. That is the reason why we selected the Argo Craft's capture card.

The tracking hardware is a dedicated hardware to calculate the cross-correlation at video rate. The hardware enables real-time correlation based tracking. It is originally designed and developed by Prof.s Inoue and Inaba of Tokyo University[8][9][10] and commercialized by Fujitsu[11][12][13]. The hardware is utilized not only for tracking but also for stereo pair matching. Skin color extraction and face identification are done with software, not utilizing any special hardware.

Auditory Processing

The prototype robot is planned to be equipped with microphones array for auditory input. The microphones array enables to form the acoustic focus mentioned above. Necessary calculation for signal processing to realize the acoustic focus is performed by Hitachi's SH series microprocessor for embedded use.

Tactile Sensation

To detect tactile information, several tactile sensors are planned to be equipped with the implementing prototype robot. The sensor to be used is FSR(Flexible Sensitive Register), a thin-film register which varies to its pressure. The sensors are planned to installed at sole, head, back and behind of the robot. Data from the tactile sensors are fed into the control computer via A/D converters.

Control Computer

As a control computer there is a onboard PC compatible small sized embedded board computer for industry use. The computer is connected with LAN via wireless ethernet adapter. Thus the robot is possible to be accessed through the network. This function is convenient and useful for development process of the prototype[14][15].

There are two reasons to choose the PC compatible as the control computer. One is wide range of hardware from small sized board to large sized rack type. And, as for the PC compatible, there has been rapid and continuous growth of hardware performance. Even

though commercially available board sized PC currently does not have so much high performance, we can expect the rapid and continuous growth and to obtain sufficient performance in the near future.

The other reason is wide range of software selection not only for applications but also operating systems and middle ware, such as data base management systems. As for the operating system, Linux OS is utilized because it is light weighted and free but true preemptive multitasking OS. Since source code of Linux OS is freely available, it is relatively easy to modify and extend to introduce experimental functions. Thus Linux OS is more suitable than other commercial operating systems for this kind of research purpose experimental system. For Linux OS there are also various practical applications and number of such kind applications has increased recently. Moreover there are practically usable and high quality middle ware, such as web server, data base management systems, and network security packages.

Software

As for software issues, there are various different requirements. They spread from milli-seconds order servo control to 10 seconds order behavior planning. Considerably large size programs, such as planning, image processing and auditory signal processing, should run concurrently. Therefore operating system for the onboard computer is demanded as 32bit multi process capability. As described above, in current implementation, we choose Linux OS as the operating system and EusLisp[16] as the base programming environment for higher level processing, such as integration of various sensation, integration of sensing and action, behavior planning, and dynamic plan revision.

4 Conclusion

This paper proposes a human following robot as a novel human collaborative robot. The robot is a legged mobile robot which is possible to follow human master properly by its vision. In this paper, the authors describe required functions to realize the human following robot and its structure to meet the requirements. Implementing issues of the prototype robot that we are now constructing are also described. Current status of the construction and preliminary study on mobility, visual processing and auditory processing are reported. Future works are completion of the construction, integration of each functions, development of software for various behavior, and experiments using the prototype robot.

References

[1] A. Pentland, "Smart Rooms", Scientific American, pp.54-62, 1996.

[2] H. Asada and I.W. Hunter, "Total Home Automation and Health Care/Elder Care", Tech. Report, Dept. of Mech. Eng., MIT, 1996.

[3] M. Fujita and K. Kageyama, "An Open Architecture for Robot Entertainment", The 1st International Conference on Autonomous Agents, pp.435-442, 1997.

[4] H. Mizoguchi, T. Sato, K. Takagi, M. Nakao, and Y. Hatamura, "Realization of Expressive Mobile Robot", Proc. of ICRA'97, pp.581-586, 1997.

[5] T. Mori, T. Kamisuwa, H. Mizoguchi, and T. Sato, "Action Recognition System based on Human Finder and Human Tracker", Proc. of IROS'97, pp.1334-1341, 1997.

[6] H. Mizoguchi, T. Shigehara, M. Teshiba, and T. Mishima, "Realizing Virtual Wireless Microphone to Pick up Human Voice Remotely and Clearly", Proc. of AROB'99, 1999. (to appear)

[7] K. Arikawa and S. Hirose, "Development of Quadruped Walking Robot TITAN-VIII", Proc. of IROS'96, pp.208-214, 1996.

[8] H. Inoue, T. Tachikawa, and M. Inaba, "Robot Vision System with a Correlation Chip for Real-time Tracking, Optical Flow and Depth Map Generation", Proc. of ICRA'92, pp.1621-1626, 1992.

[9] H. Inoue, M. Inaba, T. Mori, and T.Tachikawa, "Real-Time Robot Vision System based on Correlation Technology", Proc. of ISIR'93, pp.675-680, 1993.

[10]T. Mori, M. Inaba, and H. Inoue, "Visual Tracking based on Cooperation of Multiple Attention Regions", Proc. of ICRA'96, pp.2921-2928, 1996.

[11]T. Uchiyama, N. Sawasaki, T.Aoki, T.Morita, M. Sato, M. Inaba, and H. Inoue, "Hardware Implementation of the Video-rate Tracking Vision", Proc. of the 12th Annual Conference of the Robotics Society of Japan, pp.345-346, 1994.

[12]N. Sawasaki, T. Morita, and T. Uchiyama, "Design and Implementation of High-speed Visual Tracking System for Real-time Motion Analysis", Proc. of the 13th ICPR, pp.478-483, 1996.

[13]T. Morita, N. Sawasaki, T. Uchiyama, and M. Sato, "Color Tracking Vision", Proc. of the 14th Annual Conference of the Robotics Society of Japan, pp.279-280, 1996.

[14]H. Mizoguchi, K. Hidai, Y. Goto, M. Teshiba, T. Shigehara, and T. Mishima, "An Efficient Method to Develop Control Software of a Research Purpose Legged Mobile Robot", Proc. of '98KACC, pp.26-29, 1988.

[15]H. Mizoguchi, M. Teshiba, Y. Goto, K. Hidai, T. Shigehara, and T. Mishima, "Security Problems and Protection Methods in Remote Control Communication for Mobile Robots Using Wireless IP Network", Proc. of '98KACC, pp.401-406, 1998.

[16]T. Matsui, "Multithread object-oriented language euslisp for parallel and asynchronous programming in robotics", Workshop on Concurrent Object-based Systems, IEEE 6th Symposium on Parallel and Distributed Processing, 1994.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Internal State Acquisition for Reinforcement Learning Agent by using Radial Basis Function Neural Network

Hajime Murao
murao@al.cs.kobe-u.ac.jp

Shinzo Kitamura
kitamura@al.cs.kobe-u.ac.jp

Department of Computer and Systems Engineering,
Faculty of Engineering, Kobe University
1-1 Rokkodai, Nada, Kobe 657 JAPAN

Abstract

In this paper, we apply an adaptive Gaussian soft-max neural network to construct a state space suitable for Q-learning to accomplish tasks in continuous sensor space. In the proposed method, a state of Q-learning is defined by a hidden neuron of the neural network which is used to estimate resulting sensor signals of actions. The learning agent starts with single state covering whole sensor space and a new state is generated incrementally by adding a new hidden neuron when difference between the estimated sensor signal and incoming one exceeds a given threshold. Simulation results show that the proposed algorithm is able to construct the sensor space effectively to accomplish the task.

1 Introduction

Reinforcement learning is an efficient method to acquire adaptive behavior of a robot with little or no a priori knowledge of an environment where the robot will work. However, there is a problem in applying reinforcement learning to tasks in the real world, i.e. how to construct state spaces suitable for the reinforcement learning. A state space is usually designed by segmenting a continuous sensor space using human intuitions. Such a state space is not always appropriate to accomplish a task. Coarse segmentation will cause so-called "perceptual aliasing problem" by which a robot cannot discriminate states to be important to accomplish a task. Fine segmentation to avoid the perceptual aliasing problem will produce too many states to manage with computational resources such as CPU time and memory. It might be a rather reasonable solution to this problem applying a robot not with a state space designed by human but with a method to construct state space using information of its environment.

There were several approaches to construct a state space adaptively using sensor vectors and/or reinforcement signals. [1, 2, 3, 4, 5, 6, 7]. However, some methods based on statistical analysis of enviromental information take long time to obtain reasonable state space and some methods based on immediate reinforcement signal cannot be applied to problems for delayed reward problems.

In this paper, we propose an adaptive Gaussian soft-max neural network to construct a state space suitable for Q-learning to accomplish tasks in continuous sensor space. In the proposed method, a state of Q-learning is defined by a hidden neuron of the neural network which is used to estimate resulting sensor signals of actions. The learning agent starts with single state covering whole sensor space and a new state is generated incrementally by adding a new hidden neuron when difference between the estimated sensor signal and incoming one exceeds a given threshold. This is an online method to construct state space without a priori knowledge of environment.

The next section gives a brief review of the reinforcement learning. The adaptive Gaussian soft-max neural network is then introduced, followed by an illustration of the whole procedure of the proposed method. A result of computer simulations for a delayed reward problem is reviewed, and finally we conclude.

2 Reinforcement learning

An N_s sensors robot provides N_s-dimensional vector $\boldsymbol{s}$, for which the i-th component is a range value s_i provided by the i-th sensor. For every sensor vector $\boldsymbol{s} \in \boldsymbol{S}$, the robot can take an action a from the action set $\boldsymbol{A}$. The action $a \in \boldsymbol{A}$ for the sensor vector $\boldsymbol{s} \in \boldsymbol{S}$ causes a transition of the sensor vector to $\boldsymbol{s}' = e(\boldsymbol{s}, a) \in \boldsymbol{S}$, where e is a given transition function which defines an environment. We assume that a fitness value $v(\boldsymbol{s})$ is defined for each sensor vector $\boldsymbol{s}$. The robot however can receive reinforcement signals only, which represent partial information about fitness values of the sensor vector $\boldsymbol{s}$ and the next sensor vector $\boldsymbol{s}'$. Since the sensor vector $\boldsymbol{s}'$ is defined by a previous sensor vector $\boldsymbol{s}$ and an action a, we can define the reinforcement signal as $r(\boldsymbol{s}, a)$. A purpose of the reinforcement learning is to find a policy of selecting an action a for a sensor vector $\boldsymbol{s}$ that maximizes the discounted sum of the reinforcement signals $r(\boldsymbol{s}, a)$ received over time.

The Q-learning algorithm gives us a sophisticated solution to this problem. We assume a sensor space $\boldsymbol{S}$ is quantized into a discrete and finite set $\boldsymbol{X}$ for which a sub-region $\boldsymbol{S}^x$ of the sensor space $\boldsymbol{S}$ is characterized by a corresponding state $x \in \boldsymbol{X}$. If a sensor vector $\boldsymbol{s}$ is within this region, we say the robot is in the state x. An estimated discounted sum of the reinforcement signals $Q_n(x, a)$ for an action a is assigned to each state x, where n is the number of updates. If the robot is in a state x, an action a is selected from $\boldsymbol{A}$ according to Boltzmann distribution of a $Q_n(x, a)$ value, as follows,

$$P(a|x) = \frac{\exp(Q_n(x, a)/\tau)}{\sum_{b \in \boldsymbol{A}} \exp(Q_n(x, b)/\tau)} \qquad (1)$$

where τ is a scaling constant. In the Q-learning algorithm, the robot transits from the state x to a state $x' \in \boldsymbol{X}$ by the action a and this updates the $Q_n(x,a)$ value as,

$$Q_{n+1}(x,a) \Leftarrow (1-\alpha)Q_n(x,a) + \alpha(r(\boldsymbol{s},a) + \beta \max_{b \in \boldsymbol{A}} Q_n(x',b)) \quad (2)$$

where α is a learning rate and β is a discounting factor.

After a sufficient number of iteration, an action a which maximizes a $Q_n(x,a)$ is the optimal decision policy at a state x.

3 Adaptive Gaussian soft-max neural network

3.1 Forward dynamics

We apply here an adaptive Gaussian soft-max neural network which is a kind of the Radial Basis Function Neural Networks. This is an extension of the Gaussian soft-max network [8] previously used to estimate the actor and the critic in TD-learning.

As shown in Fig.1, the proposed neural network consists of five layers. The input layer consists of N input neurons which have no functionality but transmit input signals to neurons in the a-layer. The a-layer consists of N columns each of which contains K neurons. The output signal a_n^k of the k-th neuron in the n-th column of the a-layer is given by

$$a_n^k = \exp(-\frac{(\phi_n^k(I_n - \theta_n^k))^2}{2}), \quad (3)$$

where θ_n^k and ϕ_n^k are the center and the size of basis function of the neuron a_n^k. The output signal b_k^n of the n-th neuron in the k-th row of b-layer is defined as

$$b_k^n = \frac{a_n^k}{\sum_i^K a_n^i}. \quad (4)$$

The output signal c_k of the k-th hidden neuron and the output signal O_l of the l-th output neuron are also defined as

$$c_k = \prod_j^N b_k^j \quad (5)$$

$$O_l = \sum_k^K \omega_{kl} c_k, \quad (6)$$

where ω_{kl} is a connective strength between hidden neuron c_k and output neuron O_l.

3.2 Learning algorithm

The object of the adaptive Gaussian soft-max neural network is to minimize the squared sum of training error

$$E = \frac{1}{2}\sum_l^L (O_l - T_l)^2, \quad (7)$$

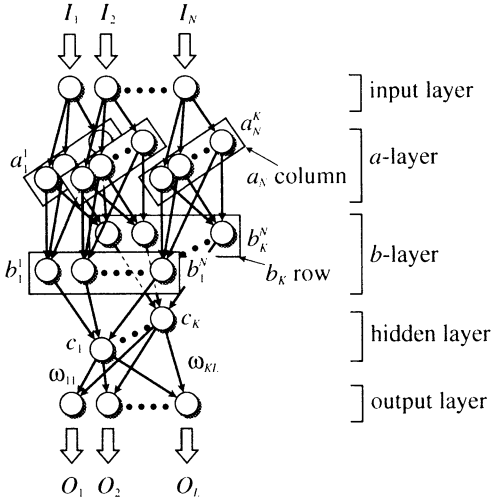


Figure 1: The structure of the adaptive Gaussian soft-max neural network.

where T_l is a teaching signal corresponding to the output neuron O_l. To minimize it, the connective strength ω_{kl} between the hidden neuron c_k and the output neuron O_l, and the center θ_n^k and the size ϕ_n^k of the a_n^k neuron are updated based on the steepest descent method with a small positive constant γ, η and ρ as follows:

$$\Delta\omega_{kl} = -\gamma\frac{\partial E}{\partial \omega_{kl}} = -\gamma(O_l - T_l)c_k \quad (8)$$

$$\Delta\phi_n^k = -\eta\frac{\partial E}{\partial \phi_n^k} = -\eta\sum_m^K\sum_l^L (O_l - T_l)w_{ml}\prod_{j\neq n}^N b_m^j \frac{\partial b_m^n}{\partial a_n^k}\frac{\partial a_n^k}{\partial \phi_n^k} \quad (9)$$

$$\Delta\theta_n^k = -\rho\frac{\partial E}{\partial \theta_n^k} = -\rho\sum_m^K\sum_l^L (O_l - T_l)w_{ml}\prod_{j\neq n}^N b_m^j \frac{\partial b_m^n}{\partial a_n^k}\frac{\partial a_n^k}{\partial \theta_n^k} \quad (10)$$

where

$$\frac{\partial b_m^n}{\partial a_n^k} = \begin{cases} \frac{1}{a_n^k}(1-b_k^n)b_k^n & \text{if } m = k \\ \frac{(b_k^m)^2}{a_n^m} & \text{if } m \neq k \end{cases} \quad (11)$$

$$\frac{\partial a_n^k}{\partial \phi_n^k} = -\phi_n^k(I_n - \theta_n^k)^2 a_n^k \quad (12)$$

$$\frac{\partial a_n^k}{\partial \theta_n^k} = (\phi_n^k)^2(I_n - \theta_n^k)a_n^k. \quad (13)$$

3.3 Adaptive increment of hidden neurons

Since the adaptive Gaussian soft-max neural network uses a radial basis function, we can add new hidden neurons without strong influence upon the output signals of the network. We define a criterion of adding a new hidden neuron based on the squred sum of training error E and the output signals of neurons in the a-layer as follows:

$$E > \varepsilon_E, \qquad a_n^k < \varepsilon_a \quad \text{for all } n \text{ and } k \qquad (14)$$

When the condition Eq. 14 is satisfied for a given threshold ε_E and ε_a, we add a new hidden neuron c_{K+1}, corresponding b_{K+1}-row and neurons a_n^{K+1} $(n = 1, ..., N)$ to the network. Connective strength ω_{K+1l}, and the center θ_n^{K+1} and the size ϕ_n^{K+1} of basis function of the neuron a_n^{K+1} are initialized to corresponding teaching signal T_l, input signal I_n and constant ϕ_0 respectively.

4 Incremental state acquisition for Q-learning

We apply the adaptive Gaussian soft-max neural network to a learning agent to estimate a resulting sensor vector of an action. Both of the number of input neurons (L) and that of output neurons (N) are fixed to the number of sensors of the agent (N_s) so as to treat an incoming sensor vector as input signals and teaching signals of the neural network. A state of Q-learning is defined by a hidden neuron of the adaptive Gaussian soft-max neural network. When the k-th hidden neuron produces the maximum output signal in the hidden layer, i.e. $c_k = \max_{i=1,...,K} c_i$ where K is the number of hidden neurons, a learning agent is said to be in a state x_k. Figure 2 shows an overview of the agent. A procedure of the proposed algorithm can be summarized as follows:

1. **[Initialization]** We start with a single hidden neuron c_1 and corresponding state x_1 covering whole sensor space $\boldsymbol{S}$. Connective strength ω_{1l}, and the center θ_n^1 and the size ϕ_n^1 of basis function of a-layer neurons are intialized to randomly chosen small numbers.

2. **[Action based on Q values]** An agent with a sensor vector $\boldsymbol{s}$ in the state x is driven according to the Boltzmann distribution described in Eq. (1) and obtains the reinforcement signal $r(\boldsymbol{s}, a)$ as a result of an action a.

3. **[Updating Q value]** If the obtained reinforcement signal $r(\boldsymbol{s}, a)$ is not equal to zero or the state has been changed as a result of the action a, the $Q_n(x, a)$ value is updated according to Eq. (2).

4. When $Q_n(x, a) = \max_{b \in \boldsymbol{A}} Q_n(x, b)$, the following procedure is done.

   (a) **[Training the adaptive Gaussian Soft-max neural network]** The adaptive Gaussian soft-max neural network is trained by using the resulting sensor vector $e(\boldsymbol{s}, a)$ as the teaching signals.

   (b) **[Adding a new state]** When the condition in Eq. (14) is satisfied, a new hidden neuron c_k and corresponding new state x_k is added. $Q_0(x_k, a)$ values for each action a is initialized to $Q_0(x_k, a) = 0$.

We note here that since the adaptive Gaussian soft-max neural network in the proposed method generates the state space independently of immediate reinforcement signals, the proposed method can be applied to problems without the immediate reinforcement signals, e.g. a problem navigating a learning agent where the reinforcement signal is given to the agent at the goal only.

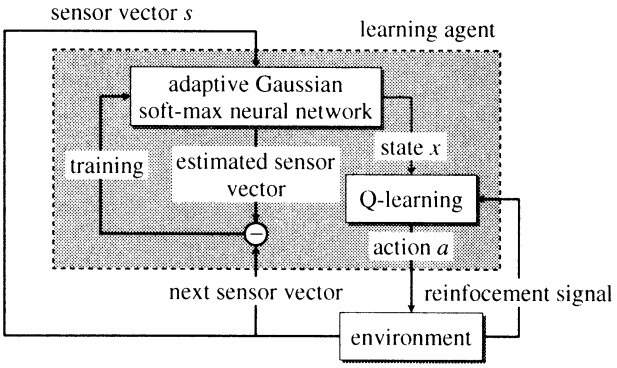


Figure 2: An overview of a learning agent.

5 Computer simulation for the problem navigating a mobile robot in a simple maze

We apply the proposed algorithm to a problem navigating a mobile robot with $N_s = 2$ sensors which provide orthogonal coordinates of the robot as $\boldsymbol{s} = (d_0, d_1)$. A task is navigating the robot from initial position to the goal in a simple maze shown in Fig. 3. In this case, we cannot define the environmental function and the fitness landscape explicitly but only define the reinforcement signal at the goal area as follows:

$$r(\boldsymbol{s}, a) = \begin{cases} 5 & \text{if the robot reaches the goal area} \\ 0 & \text{otherwise.} \end{cases} \qquad (15)$$

The action set $\boldsymbol{A}$ is defined by

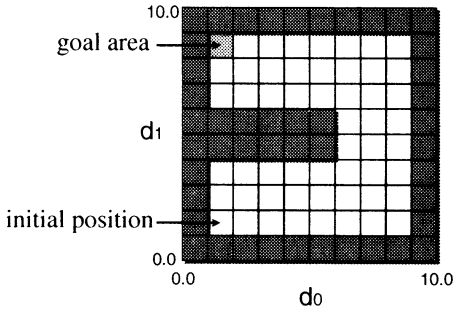


Figure 3: A simple maze for a mobile robot navigation.

$$\boldsymbol{A} = \{stop, up, down, left, right\}, \qquad (16)$$

Table 1: Parameters for simulations of navigating 2-sensors mobile robot.

parameters	description
$\tau = 1.0$	the scale constant in Eq. (1).
$\alpha = 0.5$	the learning rate in Eq. (2).
$\beta = 0.5$	the discounting factor in Eq. (2).
$\gamma = 0.1$	the small constant in Eq. (8).
$\eta = 0.1$	the small constant in Eq. (9).
$\rho = 0.01$	the small constant in Eq. (10).
$\varepsilon_E = 0.01$	the threshold for the sum-of-squared training error in Eq. (14).
$\varepsilon_a = 0.6$	the threshold for the output values of the basis functions in Eq. (14).
$\phi_0 = 4.0$	the initial size of the basis functions.

for which the environmental function $e(\boldsymbol{s}, a)$ is defined as follows:

$$e(\boldsymbol{s}, a) = \boldsymbol{s} + \boldsymbol{u}_a \tag{17}$$

where $\boldsymbol{u}_{stop} = (0,0)$, $\boldsymbol{u}_{up} = (0,0.1)$, $\boldsymbol{u}_{down} = (0,-0.1)$, $\boldsymbol{u}_{left} = (-0.1,0)$ and $\boldsymbol{u}_{right} = (0.1,0)$. Other parameters for simulations are summarized in Table 1.

A procedure of the simulation is as follows:

1. Place a robot with single state at initial position $d_0 = 1.5, d_1 = 1.5$.
2. Apply the proposed algorithm to the robot for 10000 steps till it reaches the goal. We count this 1 trial.
3. Replace the robot at randomly chosen neighborhood of the initial position and repeat trials.

As shown in Fig. 4 and Fig. 5, the number of states and the steps converge to a certain number after in early trials.

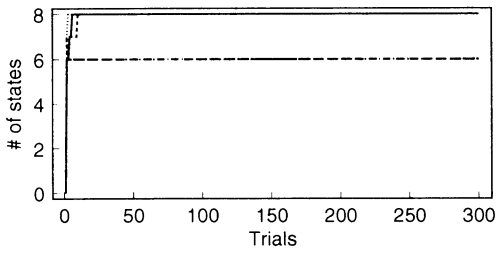


Figure 4: Time courses of the number of states for the simple maze.

6 Conclusion

We have proposed an adaptive Gaussian soft-max neural network and apply it to acquire states for Q-learning to accomplish the tasks in continuous sensor space. We think the proposed method has superiority to other approaches in some points: it needs no immediate reinforcement signals, and it generates nonlinear model of an environment by using the adaptive Gaussian soft-max neural network.

The proposed algorithm was applied to the problems navigating the mobile robot in continuous sensor space. Simulation results showed that it could generate a discrete state space efficiently to accomplish these tasks without a priori knowledge of environments.

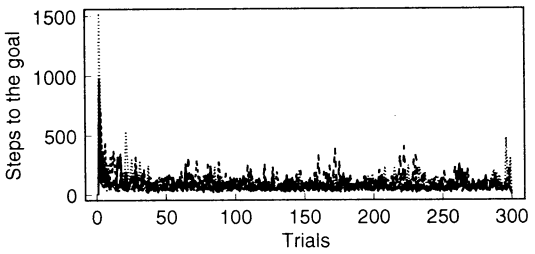


Figure 5: Time courses of the steps spent to reach the goal for the simple maze.

Acknowledgments

This research was supported by "Methodology for Emergent Synthesis" Project (project number 96P00702), "Research for the Future" Program of the Japan Society for the Promotion of Science (JSPS).

References

[1] Dubrawski, A. and Reignier, P. (1994) Learning to Categorize Perceptual Space of a Mobile Robot Using Fuzzy-ART Neural Network, *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vol.2, pp.1272-1277.

[2] Ishiguro, H., Sato, R. and Ishida, T. (1996). Robot Oriented State Space Construction, *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vol.3, pp.1496-1501.

[3] Chapman, D. and Kaebling, L. P. (1991). Input Generalisation in Delayed Reinforcement Learning: an Algorithm and Performance Comparisons, *Proc. of IJCAI-91*, pp.726-731.

[4] Munos, R. and Patinel, J. (1994). Reinforcement learning with dynamic covering of state-action space: Partitioning Q-learning, *Proc. of the Third Interational Conference on Simulation of Adaptive Behavior(SAB)*, pp.354-363.

[5] Murao, H. and Kitamura, S. (1997). Q-Learning with Adaptive State Segmentation (QLASS), *Proc. of IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, pp.179-184, Jul.10-11, 1997, Monterey, California, U.S.A.

[6] Kröse, B.J.A. and van Dam, J.W.M. (1992). Adaptive state space quantisation for reinforcement learning of collision-free navigation, *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vol.2, pp.1327-1331.

[7] Takahashi, Y., Asada, M. and Hosoda, K. (1996). Reasonable Performance in Less Learning Time by Real Robot Based on Incremental State Space Segmentation, *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vol.3, pp.1518-1524.

[8] Doya, K. (1997). Efficient nonlinear control with actor-tutor architecture, *Advances in Neural Information Processing Systems 9*, pp.1012-1018, Cambridge, MA, MIT Press.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Real-Time Search for Autonomous Mobile Robot Using the Framework of Anytime Algorithm

K. Fujisawa*1
T. Suzuki*1
*1Electrical Eng.,
Nagoya University
Furo-cho,Chikusa-ku,Nagoya,
Aichi,464-8603,Japan

S. Hayakawa*2
S.Okuma*1
*2Toyota Technological
Institute
2-12, Hisakata, Tenpaku-ku,
Nagoya, 468, Japan

T. Aoki*3
*3Nagoya Municipal
Ind., Res., Inst.
3-4-41, 6-ban, Atsuta-ku,
Nagoya, 456, Japan

Abstract

We propose a new method based on a real-time search for the optimal action of an autonomous mobile robot using the framework of Anytime Algorithm. This real time search algorithm is available in the dynamic environment. In order to apply Anytime Algorithm to the multi-objective real-time search in the dynamic situation around the autonomous mobile robot, we adopt a switching evaluation function technique, the prediction,the action watch dog and Evolution Strategy. These enable the robots to realize the real time search. This paper explains the proposed method and shows the feasibility through the experimental results using the real robots.

Keywords: Real-Time Search,Anytime Algorithm, Action Watch Dog,Switching Evaluation Function, Autonomous Mobile Robot

1 Introduction

When an autonomous mobile robot moves in the dynamic environment, it must know the pre-defined rules for actions[1] or learn the rules through the interaction with the environment[2]. Most of researches in this field have used one of them or a hybrid type [3]. Such methods need the pre-knowledge about the environment or the efficient categorization of the environment around the robot in order to learn the fine actions. In the case of using the pre-defined rules,the designer must imagine all the situations in advance where the robot faces. But it is difficult because of unpredictable situations or disturbances. On the other hand,the algorithm based on learning needs a lot of experiments to acquire the various rules. Further,it needs to effectively categorize all the situations the robot faces.

To overcome these problems,we propose a new motion planning technique based on the real-time action search under the framework of Anytime Algorithm[4] [5],which has been proposed in the area of Artificial Intelligence. Anytime Algorithm iteratively improves a quality of result over calculation time and it is an interruptible algorithm at anytime. It can also allows subsystems to intelligently allocate computational time resources.

Using this idea,the robot can adapt the dynamical environment. Moreover, designers are relieved from the problem of the pre-defined rules and the pre-classification of the situations. The key point of the proposed method is the trade-off between an amount of calculation and a quality of solution because the robot has to do the recognition of the environment,exploration of the optimal action and execution within the limited time.In the safe situation,the robot have enough time to search the optimal action.On the contrast, the robot has to search a certain action in very short time in the dangerous situation even if the result does not have good quality. Anytime Algorithm can harmonize such a trade-off as mentioned above in the statistic situation. But we cannot directly introduce Anytime Algorithm because we aim to realize the multi-objective real-time search in the dynamic environment. So,we expand the framework of Anytime Algorithm to the action search in the dynamic environment,and apply it to the motion planning for the autonomous mobile robot. To realize the expanded Anytime Algorithm,we use Evolution Strategy(ES) [6],Action Watch Dog,Prediction and a Switching Evaluation Function Technique.By using them, our proposed method can stop the action search at anytime and have an executable solution. We apply the proposed real-time search method to the moving obstacle avoidance problem. We test our proposed method using the real robots and show the feasibility.

In the following sections, the details of the proposed method and the structure of the system are explained.

2 Framework of Problem

There are diverse studies using autonomous mobile robots. In this paper, the proposed method is applied to motion planning(action searching) of an autonomous mobile robot avoiding moving obstacles. Before describing the details of our method, the following conditions are assumed.

1. The robot avoids obstacles with both steering and velocity control inputs, simultaneously.
2. There are only moving obstacles in the environments around the robot.
3. The robot with two wheels can move to all the directions without limit of paths.
4. The goal of the robot is given in advance.
5. The robot and obstacles move smoothly.

Under the above conditions, the system realizing the proposed method is constructed. The main concept of the method and the system are described in Section 3 and 4. The experimental results and the investigation are explained in Section 5.

3 Real-Time Search and Expansion of Anytime Algorithm

3.1 Real-Time Search

3.1.1 Definition of Real-Time

The term "real-time" used in this paper should be defined. The definition is highly related to this application. The term "real time" is defined as time length that there is little difference in the actual situations between calculating and executing the proper action. In the dynamical environment, the situation the robot faces is always changing as shown in Figure1. The calculated proper action may become improper in the situation where the robot executes it. In other words, the robot should sample the situation, calculate and execute the proper action in a much faster rate than the changing rate of the environment. If the situation is safe and changes slowly, the robot has enough time to search the proper action and the result is available to the current situation. If it changes fast, however, the robot dose not have enough calculation time and the sampling rate can not be up unlimitedly. To cover the situation gap and the problem in sampling rate, we introduce prediction into the system.

3.1.2 Prediction

To cover the difference between situations at the time of recognition and execution shown in Figure1 as well as possible, prediction is introduced in this system. The basic idea is that the robot predicts the future situation where the robot will execute the action and that it calculates the proper action according to the

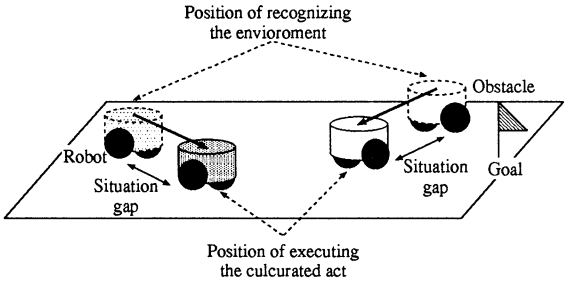


Figure 1: Situation Gap

predicted situation. Using this prediction, the robot calculates the proper action in the virtual environment. The length of the each time step is changeable and decided by the facing situations. If it is safe for a while, the interval is long. But, it is short when the situation is dangerous.

3.2 Expansion of Anytime Algorithm

3.2.1 Anytime Algorithm

Anytime Algorithm is the real time search algorithm which has been proposed in the field of the artificial intelligence. It iteratively improves a quality of result over calculation time and it is an interruptible algorithm in the middle of producing results. It can also allows sub-systems to intelligently allocate computational time resources efficiently. But it can't be introduced to the real time search in the dynamic situation. This is because of the following reasons.

We show the property of Anytime Algorithm.

1. Quality Measure
2. Predictability
3. Interruptibility and Continuation
4. Monotonicity

In the dynamic situation around the robot,it is impossible to have stational information about the output quality given a certain amount of time, because the environment is always changing. It means the property2 can't be ensured. Moreover,it is difficult to ensure the property4 to all specs in the multi-objective action search problem. Because of this,we can't directly introduce Anytime Algorithm.

So,we define the property of Anytime Algorithm in the dynamic environment as follows.

1. Quality Measure
2. Interruptibility and Continuation
3. Search based on the priority

To ensure the above property,we use the Evolution Strategy(ES) as a search method.Moreover,and adopt the Action Watch Dog and the Switching the Evaluation Function method. These are explained in the following subsections.

3.2.2 Action Watch Dog

As shown in the subsection 3.1.2,the action search is performed in the predicted situation. The prediction system has some problems. The prediction is not always the same as the real situation. When the obstacles perform the unpredictable action, the prediction error is occurred. The prediction error may cause crashing obstacles. In a such case,the search algorithm interrupts the search and re-allocates the time for the search according to the property of Anytime Algorithm. To ensure this property,we adopt the Action Watch Dog. The algorithm and the time chart are shown in Figure 2. As shown in the Fig.2,it watches the actual situation in parallel during the action search. When the dog detects the difference between prediction and real situation, it interrupts the calculation of the action search and makes the robot avoid the obstacles according to the solution.

By using this mechanism, the robot can avoid the dangerous situation even if the prediction error is occurred by unpredictable disturbance and/or change of situation.

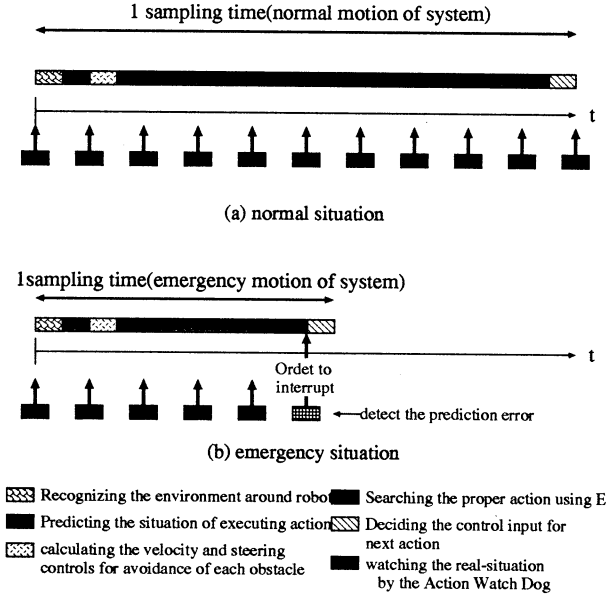


Figure 2: system's time chart

3.3 Switching the Evaluation Function

In the previous subsection,we represented about the Action Watch Dog. It interrupts the search and calculates the control input according to the solution in the middle of search when there is the difference between the real situation and the predicted situation. In this case,the solution may be converged well. It is the same when the robot can't have much calculation time because of the fast changing rate of the environment. In the case of using Anytime Algorithm in the statics environment, it is possible to guess which the solution is converged well or not. But in the dynamic environment,it is impossible because the situation is always changing and the robot can't have the statistic information about the quality of the solution to the given amount of time. In this case, it is desirable that we can ensure at least safety even if the robot uses such solution. In order to meet this requirement we adopt a Switching the Evaluation Function technique. It has some function criteria based on the priority of the required spec to the problem. In the moving obstacles avoidance problem,it has two spec. One is the avoiding obstacles.The other is the reaching to the goal. Then,we set the evaluation function as follows.

Evaluation Function

1. a function based on only the point of the avoidance-oriented standard.
2. a function based on both points of the avoidance-oriented and the goal-oriented standards.

Now,we call this function a Two-Stage Evaluation function. At first,the robot searches the proper action based on the first evaluation function.In other words,during the first stage of action search, the search algorithm calculates the evaluation signal from only a point of safety.After the first stage, the evaluation is done from both points of safety and distance to a goal. By using this mechanism, even if the search process is interrupted by unpredictable disturbance and/or change of situation, the robot can be expected to execute at least safe motion.

4 Motion Planning System for Moving Obstacles Avoidance

In this section, we explain the motion planning system for moving obstacles avoidance which we constructed and its flow. The Figure 3 shows the system. It mainly consists of three modules which are Sensor Module,Fuzzy Rules and Neural Network Module, and Virtual Space.

The system's flow and each module's function are follows.

System's Flow

1. Sensor

   In this module,the robot recognizes the current situation around itself, and predicts the situation where it performs the action.

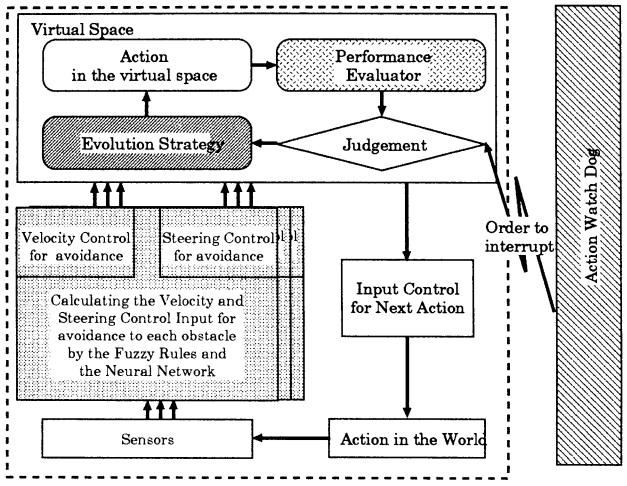


Figure 3: System of Motion Planning for Mobile Obstacle Avoidance

2. Fuzzy Rules and Neural Network

   Using the information obtained in the sensor module,the robot calculates the velocity and steering control inputs for avoiding an each obstacles by the fuzzy rules and the neural network.

3. Virtual Space

   In this module,the robot searches and decides the proper weights on the velocity and steering control inputs which are obtained in the previous module. This module consists of three submodules as follows.

   (a) Evolution Strategy(ES)

   This module searches the proper weights w_{sk} and $w_{vk}(k=1,2,3....)$ in Equation (1),(2). $w_{sk}(k=1,2,3....)$ denotes the weight on the steering control input for the kth obstacle. w_{vk} is the one on the velocity control input.

$$V_{in} = V_{now} + \sum_{r=1}^{k} w_{vk} \cdot vd[r] \quad (1)$$

$$S_{in} = rta(1 - w_s) + \sum_{r=1}^{k} w_{sk} \cdot ster[r]/k \quad (2)$$

V_{in} :velocity control input to robot
S_{in} :steering control input to robot
V_{now} :present velocity
$vd[r]$:velocity control input for avoidance
$ster[r]$:steering control input for avoidance
rta :relative angle to goal
w_{vk} :weight against $vd[r]$
w_{sk} :weight against $ster[r]$
w_s :total of w_{s1}
k :obstacle's number

   (b) Performance Evaluation

   In this module,the results of the performance based on the weights which are decided in the ES module are evaluated. In the evaluation,the two-stage evaluation function is used.

   (c) Judgement

   This module judges which the order to interrupt the search is given by the Action Watch Dog, or which the action search finishes.

5 Experimental Results and Verification

In this section, the experimental setup is shown and the results are explained to confirm the feasibility of our proposed method.

5.1 Experimental Setup

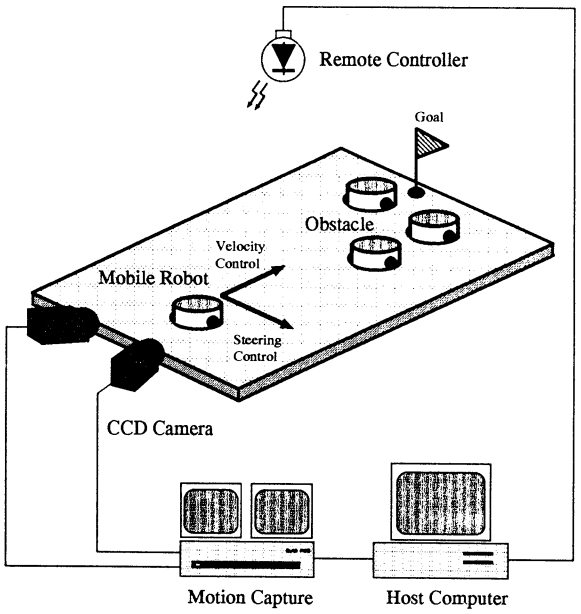


Figure 4: Experimental Setup

The assumptions made in the experiments are as follows.

- Obstacle

  1. The number of obstacles is three.
  2. The start points are set at random in the front side of the robot at every run.
  3. The moving directions are set so as to approach the robot

4. The Obstacles try to move straight..

- Robot.
  1. The start point, moving direction and initial velocity is same at every run.
  2. The robot avoids obstacles using both steering and velocity control inputs, simultaneously.

Figure 4 shows the experimental setup. The mobile robot and three obstacles used in this experiment are Khepera made by EPFL. The robot is controlled by a Fujitsu DeskPower FMM200 Computer (CPU:Intel Pentium 200MHz) via a IR controller. This controller broadcasts the command from the computer on the air. The Khepera catches the command using light-received sensor. As for the sensor to recognize the situation around the robot, two CCD camera is used. The camera is set at the side of the experiment field to overview the area. The camera is also connected to the 3D motion picture processing unit. The positions of the Kheperas are sent to the computer in the video rate. This period is same as that of the Watch Dog. All processes such as picture reading and analyzing, decision making and motor control are performed with sampling rate 0.7 sec. The experimental system is depicted in Figure 4. Then using this system, the feasibility of the proposed method is tested.

5.2 Experimental results

Figure 5 and 6 show the experimental results. In these figures, the circles marked black indicate the positions of the robot and obstacles at the same time.

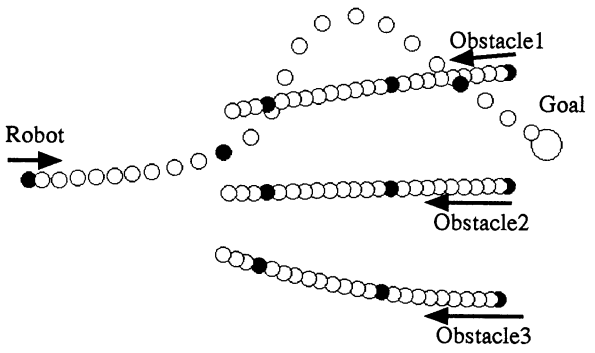


Figure 5: Experimental Result No.1

Figure 5 shows the case where the obstacles approach the robot from the front side. In this case, the watch dog dose not interrupt the action search. As

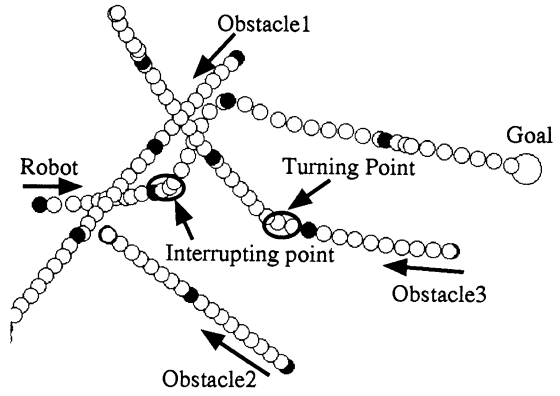


Figure 6: Experimental Result No.2

shown in this figure, the robot accelerates after detecting dangerous obstacles. Then, it turns large, moves around Obstacle 1 and gets to the goal.

Figure 6 shows the case where one of the obstacles suddenly change the moving direction. At the early steps,the robot avoids the two obstacles approaching from both side by acceleration and slight left turn. After that, another obstacle suddenly turns to the right against prediction. At the time, the Watch Dog interrupts the action search and makes the robot to ensure the safety.Then, the robot avoids the obstacle and moves to the goal.

As shown these results,the robot can acquire the fine action by recognizing the current situation, searching and executing the action in real time. Further,the Watch Dog interrupts the action search before convergence because of the unpredictable motion of the Obstacle3, and the robot executes the solution based on the first-stage evaluation of the Two-Stage Evaluation. By using this mechanism, even if the search process is interrupted by unpredictable disturbance and/or change of situation, the robot can execute at least safe motion to ensure the safety.

Because of limit of computational power and hardware, the proposed method has used the Two-Stage Evaluation technique. If possible,we want to try the multi-stage evaluation technique. According to the priority of the evaluation standard,the optimal solution can be searched by gradually narrowing the search space. In this experiment, the robot watches just the front between -100 and 100 degrees and the obstacles move forward . It is needed to discuss the relation between the sight range of the robot and the motion direction of the obstacles. Moreover, the proposed method needs to apply to the case of several or many robots. When each robot has its own action

strategy, a meta module should be considered to keep the society and emerge a new macro rules.

5.3 Verification of Two-Stage Evaluation

In this subsection, we indicate the verification of Two-Stage Evaluation using simulation data because it is difficult to get experimental data. Figure 7 shows the time chart of the two parameters which are the change of the distance to goal and the danger degree to obstacles using proposed method. This simulation was carried out using the obstacles' trajectories including the obstacle which suddenly turned at the 10^{th} step.

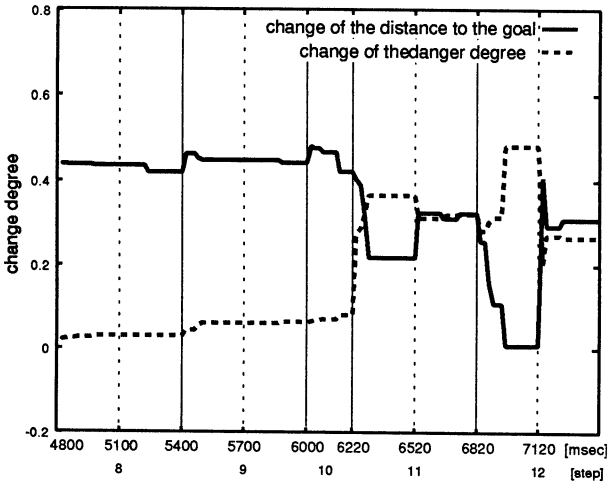


Figure 7: Time chart of change of distance to goal and danger degree (2stage evaluation with the action watch dog)

In the figure, the horizontal axis indicates the time step from the 8^{th} to 12^{th}. The vertical thick lines express the border of each step. One step is basically 600ms, which is also the sampling time(control period) without interruption. The 10^{th} step is narrower because the Watch Dog detected the prediction error and interrupted searching the proper action. The vertical dotted line in the figure indicates the switching the evaluation criteria from the first stage to the second stage. The longitudinal axis shows the change degree.The thick line indicates the change of the distance to the goal and the dotted line shows the change of the danger degree to obstacles. The more positive these parameters are, the less the distance and the danger degree become.

In this figure at early steps, the thick line is high-valued and the dotted line is low-leveled. It means that the robot selects just the goal-oriented action because the robot is in the safe situation. After interrupting the search at the 10^{th} step,at the first stage of the 11^{th} and 12^{th} steps, the dotted line is high-valued and the thick line is low-leveled. It means that the robot selects just the avoidance-oriented action because the robot is in the dangerous situation. Then,at the second stage of each step,the robot searches the goal-oriented action starting from the first stage result. The dotted line is down and the thick line is up.

In this way,by using the Two-Stage Evaluation, the robot can keep the safe situation at least, even if the action search is interrupted and the solution is not converged well. By this result, We showed the efficiency of the Two-Stage Evaluation.

6 Conclusions

We propose a new method of the real-time action search for an autonomous mobile robot using the framework of Anytime Algorithm. By using our proposed method, the robot does not need to classify the all of the situation in advance,or to efficiently categorize the situation to learn the action rules. For the future works, we will adopt learning system in the proposed system. To memorize the calculated action, the robot use the computational resources, efficiently.

References

[1] Y.Maeda et.al: "Avoidance Control among Moving Other robots for a Mobile Robot on the Fuzzy Reasoning",J.of the Robotics Society of Japan,Vol.6,No.6,pp.518-522,1988

[2] T.Tsubouchi et.al: "A navigation scheme with learning for a mobile robot among multiple moving obstacle",1993 IEEE International Conference on Intelligent Robots and Systems, Yokohama in Japan, July26-30,pp.2234-2240

[3] T.Aoki et.al: "Acquisition of Optimal Action Selection to Avoid Moving Obstacles in Autonomous Mobile Robot", Proc. of IEEE International Conference on Robotics and Automation,Minneapolice in U.S.A,Apr.22-28, pp.2055-2060, 1996

[4] S.Zilberstein: "Using Anytime Algorithms in Intelligent Systems", AI Magazine,17(3),pp.73-83,1996

[5] J.Grass: "Reasoning about Computational Resource Allocation -An introduction to anytime algorithms",the ACM magazine,1996

[6] Ralf Salomon: "Increasing Adaptivity through Evolution Strategies", The Fourth International Conference on Simulation of Adaptive Behavior,1996

[7] Ferrari.C et.al: "Varying Paths and Motion Profiles in Multiple Robot Motion Planning", Proc. Int.Symp.Computational Intelligence in Robotics and Automation, pp.186-193,1997

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Incremental Evolution of CAM-Brain to Control a Mobile Robot*

Geum-Beom Song and Sung-Bae Cho

Department of Computer Science, Yonsei University
134 Shinchon-dong, Sudaemoon-ku, Seoul 120-749, Korea
Phone: +82-2-361-2720, Fax: +82-2-365-2579
E-mail: [goldtiger, sbcho]@candy.yonsei.ac.kr

Abstracts

There has been a significant progress in research to control mobile robots in simulation and real world. Several researchers have attempted to construct the mobile robot controller that can avoid obstacles, evade predators, or catch moving prey by evolutionary algorithms such as genetic algorithm. In previous research, we presented the CAM-Brain that evolved neural networks based on cellular automata (CA) and applied it to control a mobile robot for showing the usefulness. However, this direct evolution has a difficulty that cannot generalize the controller well to new environments. This paper attempts to solve it by incremental evolution, which starts with simpler environments and gradually develops the controller with more general and complex environments. Experimental results show that the incremental evolution develops the robot controller efficiently and robustly.

Keywords

CAM-Brain, Cellular Automata, Neural Networks, Mobile Robot Control, Incremental Evolution.

1. Introduction

There are many studies of constructing mobile robot controller by evolutionary approaches such as evolving neural network by genetic algorithm [1], using genetic programming [2], and combining fuzzy controller with genetic algorithm [3]. Usually, genetic algorithm evolves successful robot controller but this way tends to produce controller constrained in a given environment.

In previous work [4], we introduced CAM-Brain, evolved neural networks based on cellular automata [5], which can perform complex behavior by combining simple rules. Also, we applied it to controlling a mobile robot for showing the usefulness. However, the controller obtained had a difficulty to adapt in changing environment.

In this paper, we attempt to devise a sophisticated method based on incremental evolution for solving this problem. Incremental evolution does not evolve controller directly to do goal behavior in an environment, but starting with simpler environments gradually develops the controller with more general and complex environments [6]. We expect that this way makes complex and general behavior which can adapt in changing environment.

In this paper, we develope a mobile robot controller which navigates different environments without bumping against obstacles by evolving a CAM-Brain module in the incremental fashion. Because a mobile robot can perform complicated behaviors with the combination of going straight and turning left and right, we attempt to evolve it to do these basic behaviors by incremental evolution and show the simulation results. The usefulness of the proposed method will be presented by applying the evolved CAM-Brain module to more complex and general environments that are different from those used in evolution process.

2. Neural Networks based on CA

CAM-Brain is a model based on CA which can show complicated behavior by combining simple rules, and developed by its own chromosome that has information about CA-cell structure. One chromosome is mapped to exactly one neural network module. Therefore, with genetic algorithm working on this chromosome, it is possible to evolve and adapt the structure of the neural network for a specific task. It is the basic idea of CAM-Brain that brain-like system can be made by combining many neural network modules that have various functions [5]. This section illustrates a design of CA-space for

* This work has been supported in part by a grant from the Ministry of Science and Technology in Korea.

developing a neural network module.

2.1 CoDi Model

CAM-Brain's neural network structure composed of blank, neuron, axon and dendrite is grown inside 2-D or 3-D CA-space by state, neighborhoods and rules encoded by chromosome. If cell state is blank, it represents empty space and cannot transmit any signals. Neuron cell collects signals from surrounding dendrite cells which are accumulated. If the sum of collected signals is greater than threshold, neuron cells send them to surrounding axon cells. Axon cell sends signals received from neurons to the neighborhood cells. Dendrite cell collects signals from neighborhood cells and passes them to the connected neuron in the end [5].

2.2 Growth Phase

The growth phase organizes neural structure and makes the signal trails among neurons. First, a chromosome is randomly made and the states of all cells are initialized as blank. At this point some of the cells are specified as neuron with some probability. Then, a neuron cell sends axon and dendrite growth signals to the direction decided by chromosome. Axon growth signal is sent to two directions and dendrite growth signal is sent to the other remaining directions. Next the blank cell received growth signal changes to axon or dendrite cell according to the type of growth signal. It sends the signals received from other cells to the direction determined by chromosome. Finally, repeating this process, the neural network is obtained when the state of every cell changes no longer.

Fig. 1 shows the growing process in 4×4 2-D CA-space. In this figure, the cell which has oblique lines is blank cell, and the black arrows show the direction of signaling decided by chromosome. Fig. 1 (a) shows the process of seeding a neuron in blank cells, where a neuron is located in (x_2,y_2). Fig. 1 (b) shows that the neuron cell sends growth signal to surrounding cells. Fig. 1 (c) shows the cell state is changed by growth signal. Fig. 1 (d) shows that blank cells grow into axon or dendrite. In a neuron, the dendrite collects signals and sends to the neuron, and the axon distributes signals originated from the neuron.

2.3 Signaling Phase

Signaling phase transmits the signal from input to output cells continuously. The trails of signaling are transmitted with evolved structure at the growth phase. Each cell plays a different role according to the type of cells. If the cell type is neuron, it gets the signal from connected dendrite cells and gives the signal to neighborhood axon cells when the sum of signals is greater than threshold. If the cell type is dendrite, it collects data from the faced cells and eventually passes them to the neuron body. If the cell type is axon, it distributes data originating from the neuron body.

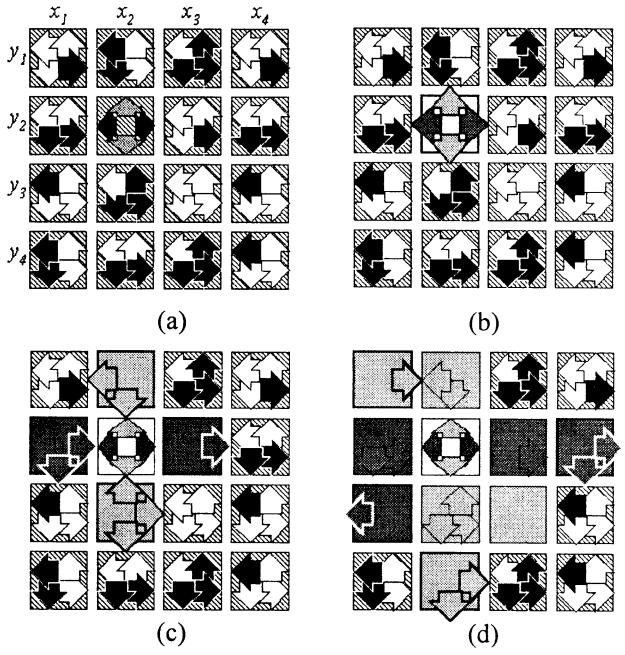


Fig. 1 Growth phase. (a) Black arrow represents signaling direction determined by chromosome, and a neuron is located in (x_2,y_2). (b) The neuron sends growth signals. (c) The cell state is decided according to the type of growth signals. (d) Propagating growth signals, blank cells become axon or dendrite.

The position of input and output cells in CA-space is decided in advance. At first, if input cells produce the signal, it is sent to the faced axon cells, which distribute that signal. Then, neighborhood dendrite cells belonged to other neurons collect and send this signal to the connected neurons. The neurons that have received the signal from dendrite cells send it to axon cells. Finally, dendrite cells of output neuron receive and send this signal to the output neurons. Output value can be obtained from output neurons. During signaling phase, the fitness is evaluated by the output in this process. Fig. 2 shows the process of signaling after neuron, axon and

dendrite are made.

2.4 Evolution of CAM-Brain

In general, simple genetic algorithm generates the population of individuals and evolves them with genetic operators such as selection, mutation, and crossover [7]. We have used the genetic algorithm to search the optimal neural network. At first, a half of the population that has better fitness value is selected to produce new population. Two individuals in the new population are randomly selected and parts of them are exchanged by one-point crossover. The crossover is occurred at the same position in the chromosomes to maintain the same length in chromosomes. Mutation is operated in the segment of chromosome. The genetic algorithm generates a new population from the fittest individuals on the given problem.

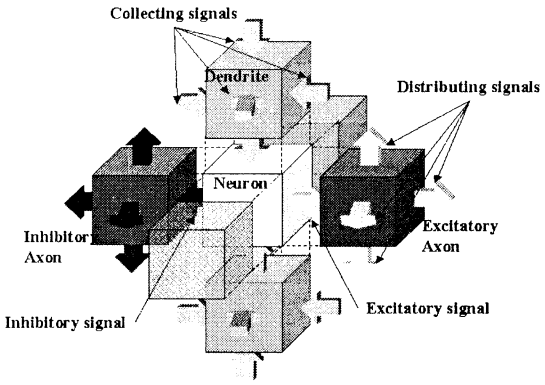


Fig 2. Signaling phase.

3. Incremental Evolution

3.1 Applying CAM-Brain to Khepera

Khepera robot contains 8 infrared sensors to detect by reflection the proximity of objects in front of it, behind it, and to the right and the left sides of it, and to measure the level of ambient light all around the robot. Also, the robot has two motors to control the left and right wheels. Khepera simulator also features the ability to drive a real Khepera robot, so that we can very easily transfer the simulation results to the real robot [2, 3, 4].

3.2 Simulation Environment

We use $5 \times 5 \times 5$ CA-space that solves the problem. Only four sensors are used in this simulation and each input cell is in the center region of the four faces of hexahedrons CA-space. Output cells are located in the top and bottom faces of hexahedrons. One robot completes the growth phase with a chromosome and then starts receiving inputs. Only neuron cells can take input and signaling phase is performed for some steps per one sensor sampling time unit.

Fitness is evaluated based on the distance between the present location of the robot and the goal point, and the number of movement going straight.

$$Fitness = (50 / D) * (\frac{1}{S} \sum_{i=0}^{S} V_i) \quad (1)$$

D: Distance between the robot and goal position
S: The number of movement until stopping
$V_i = a$(the value decided by the velocity in ith step)

The fitness evaluation leads the robot to arrive at goal position quickly without bumping against walls. Population size is 50 and the fitness of individual is computed by the average of four experiments.

The robot controller is evolved incrementally by starting with simpler environments and gradually evolving the controller with more general and complex environments. The environments get more sophisticated from straight movement to left and right turn movements. Consequently, the robot controller that can move straight and turn left and right can be obtained.

After the CAM-Brain module is evolved in the environment intended to go straight, successful chromosomes are copied to the next population. Then it is evolved in the environment intended to go straight and turn right. Progressing this process the controller evolves to go straight and turn left and right. Efficient evolution is expected because of the reduced search space by incremental evolution [8].

3.3 Results and Analysis

Fig. 3 shows the trajectories of a successful robot in each environment. The environment in Fig. 3 (a) leads the robot to going straight on the situation of no obstacles and Fig. 3 (b) does in the situation of obstacles. Fig. 3 (c) and (d) induce to turn right and Fig. 4 (e) and (f) induce to turn left. Because the robot shows the difficulty to evolve in the environment with the corridors bent radically, the environments for turning right (Fig. 3 (c) and (d)) and left (Fig. 3 (e) and (f)) are provided in two steps.

Fig. 4 shows the change of fitness in each environment shown in Fig. 3. The 1st, the 6th, the 8th, the 10th, and the 11th generations correspond to

the best fitness for each environment in Fig. 3(a), (b), (c), (d), and (e), respectively. The remaining generations show the change of fitness for the environment in Fig. 3 (f). In this figure, direct evolution means the change of the best fitness when the controller is evolved only in Fig. 3 (f) environment. This indicates that the incremental evolution method produces successful controller, while the direct evolution method cannot. It is a very promising result that the incremental evolution is more efficient than the usual direct evolution.

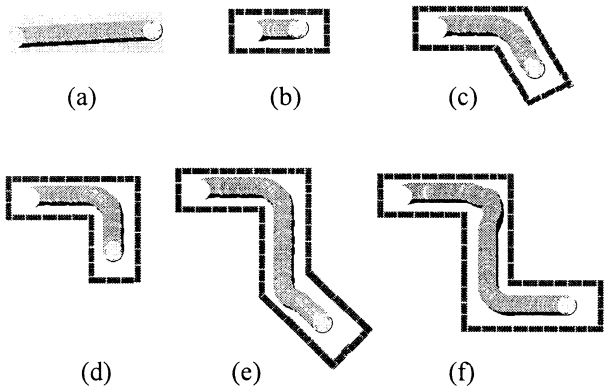


Fig. 3 Trajectories of the successful robot in each environment.

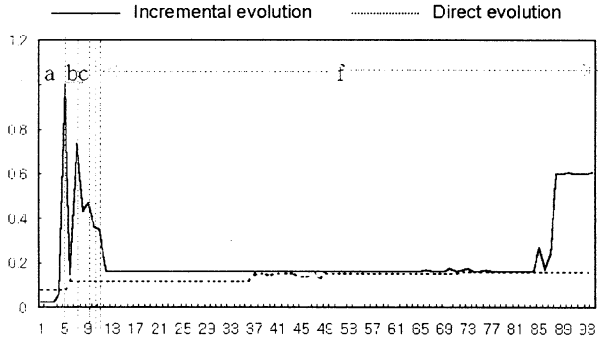


Fig 4. Comparison of the change of best fitness.

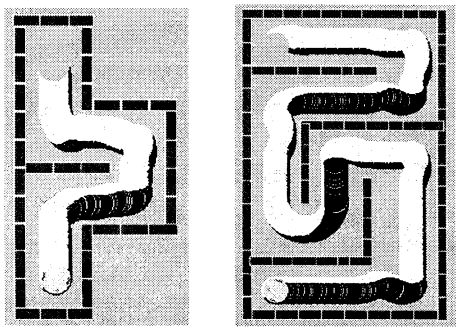


Fig. 5 Applying to different environments.

Fig. 5 shows the results of applying the evolved controller to different and more difficult environments. The best individual evolved by the incremental evolution can navigate smoothly in the new environments required the behaviors composed of going straight and turning left and right.

4. Concluding Remarks

In this paper, we have attempted to evolve a CAM-Brain module to control a mobile robot in more complex environments by incremental evolution for studying on efficient learning method. Experimental results have showed the incremental evolution can evolve CAM-Brain module more efficiently than direct evolution, and the controller evolved by this method adapts to new environments. The controller have evolved from simple to complex environments step by step, but the number of generations needed to work out each environment varies significantly. This indicates the environments decided by incremental evolution should be designed efficiently.

Reference

[1] D. Floreano and F. Mondada, "Evolution of homing navigation in a real mobile robot," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. 26, No. 3, pp. 396-407, June 1996.

[2] P. Nordin and W. Banzhaf, "Real time control of a Khepera robot using genetic programming," *Cybernetics and Control*, Vol. 26, No. 3, pp. 533-561, 1997.

[3] S. B. Cho and S. I. Lee, "Evolutionary learning of fuzzy controller for a mobile robot," *Proc. Int. Conf. on Soft Computing*, pp. 745-748, Iizuka, Japan, 1996.

[4] S. B. Cho, J. H. Lee, G. B. Song and S. I. Lee, "Evolving CAM-Brain to control a mobile robot," *Proc. AROB 98*, pp. 271-274, Beppu, Japan, January 1998.

[5] F. Gers, H. de Garis and M. Korkin, "CoDi-1Bit: A simplified cellular automata based neuron model," *Proc. Artificial Evolution Conf.* Nimes, France, October 1997.

[6] F. Gomez and R. Miikkulainen, "Incremental evolution of complex general behavior," *Adaptive Behavior,* Vol. 5 Issue 3-4, pp 317-342, 1997.

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, 1989.

[8] I. Harvey, P. Husbands and D. Cliff, "Seeing the light: Artificial evolution, real vision," *Proc. of 3rd Int. Conf. on Simulation of Adaptive Behavior,* pp. 392-401, MIT Press/Bradford Books, 1994.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Path Planning for Mobile Robot Using A Genetic Algorithm

Satoshi Tamura, Makoto Takuno, Toshiharu Hatanaka and Katsuji Uosaki
Department of Information and Knowledge Engineering
Tottori University
Tottori, 680-8552, Japan

Abstract

In this paper, an optimal path planning problem for the auto-mobile robots with four independent arms is considered. We propose a path planning procedure by using the genetic algorithms from the reference path given by the experienced operator. Computer simulation results show the validity of the proposed procedure.
Keyword: path planning, genetic algorithm, mobile robots, reference path

1 introduction

An optimal path planning or navigation problem for the auto-mobile robot has been studied in recent years. This problem is summarized as follows [1] [2]:*Given a robot and a description of an enviroment. plan a path between two specified locations, generally these locations are start and goal points, which is collision-free and satisfies certain optimization criteria.*

Though most approaches for path planning can find the optimal path under some criterion on the path, the computation burden generally becomes huge for the case of big operational space and no constraints imposed on the paths. On the other hand, it is known that experienced operators can heuristically give good path of the mobile robots for given operational space. Hence, we can expect that we can find the optimal path more easily by updating the path given by the experienced operators. From this point of view, we propose a path planning procedure, which finds a path of mobile robots with better performance by modifying the reference path given by the experienced operator. The modification is carried out by using the genetic algorithms(GA). The computer simulation results illustrate the usefulness of the proposed approach.

In Section 2, the specifications of mobile robots and operational space are presented, and the proposed modification procedure by using the GA is given in Section 3. Computer simulation results illustrating the performance of the proposed approach are presented in Section 4.

2 Description of robot and enviroment

We consider a auto-mobile robot with four independent arms illustrated in Fig.1. The robot can move on the operational space along an operational path and explores the points on the operational space using one of the arms. The other arms are used to hold the position of the robot. The feasible region for each arm is also illustrated in Fig.1. The robot explores the points in this region in order according to the control sequence shown in Fig.2. Then, if at least one of the

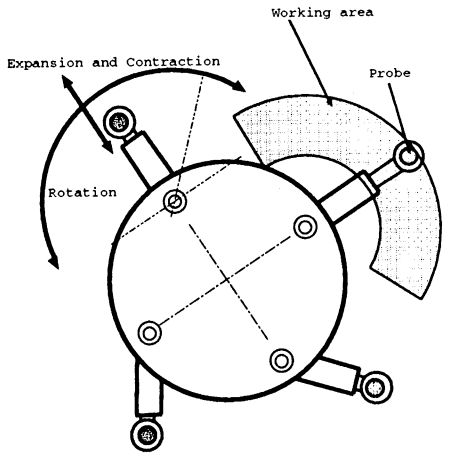


Figure 1: 4-armed mobile robot model

arms finished its task, the robot moves to a new operational location on the operational path. The landscape of operational space is given for each operation. Its shape is different for each operation. Even for operational spaces with same shape, the operational path may be different for each operation since some of the

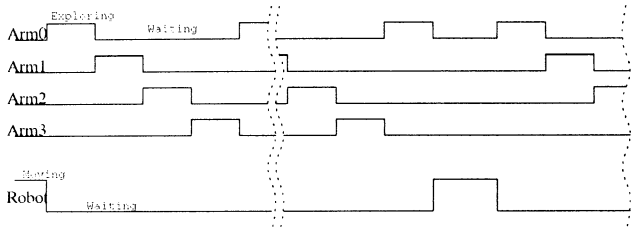


Figure 2: Control sequences of arms

points are not necessary to be explored, which are depending on the given operation conditions.
In this case, the mobile robot should explore all the points in the operational space and the optimal operational path and operational locations, which explore all the points in the operational space, are determined before the operations.

3 Modification procedure of the reference path

Since it is required to explore all the points in shorter time, the path of robot should be planned suitably.
Takuno et.al. proposed the path planning method based on the one–step optimization and shows its efficiency by some numerical examples [1]. Though their method provides a good solution, the given path is somewhat complex one and the computation burden to find the path is big.
While experienced operators can heuristically give a good but not optimal path based on their experiences and knowledge. This fact leads the possibility to improve the reference path, which is provided by operator, under the condition that the reference path should not be much deformed. We assume that the reference path consists of the start, goal points and several turning points shown in Fig.3. As noted above, we have to determine the optimal operational path and operational locations for the mobile robot.. Here, we first determine the optimal operational path, and then determine the operational locations on the determined path.
To find the operational path, we apply the genetic algorithms.

Encoding of the paths as strings
The operational path is expressed by the locations of

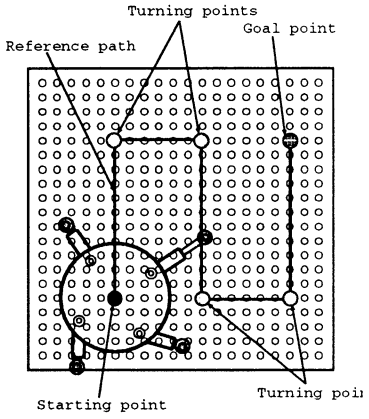


Figure 3: operational space and the reference path

the start, goal, and turning points. We put an integer representing a point, which modifies each of the start, goal, and turning points, as shown in Fig. 3. Then, the chromosome is composed by the string of integers and represents a modified path.

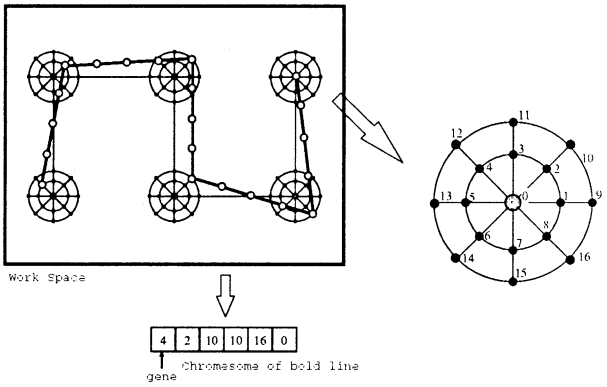


Figure 4: Genetic code for path planning

Fitness function
The exploring rate defined by the number of explored points/number of points on the operational space for the operational path represented by the chromosome is employed here. Since the rank selection rule is employed in the following, the path length is the second fitness function when the exploring rate is same.
Then the genetic algorithm is applied as follows:

Step 1: Generate the modified paths as chromosomes in the population.

Step 2: Evaluate the fitness function for the modified

paths based on the computer simulation.

Step 3: Select the parents based on the rank selection rule with the elitist strategy and produce the offsprings with mutation and crossover.

Step 4: Repeat Steps 1 to 3 until the stopping condition is satisfied.

Then, the operational locations should be determined. First, we allocate sufficiently many operational location candidates with equi-distance on the operational path. We find the operational locations by excluding unnecessary operational locations among the candidates. The selection will be done by applying the genetic algorithm.

Encoding of the locations as strings

We put an integer for each operational location candidate. The chromosome is composed by a string of integers randomly chosen from these integers and arranged in order, as shown in Fig.5.

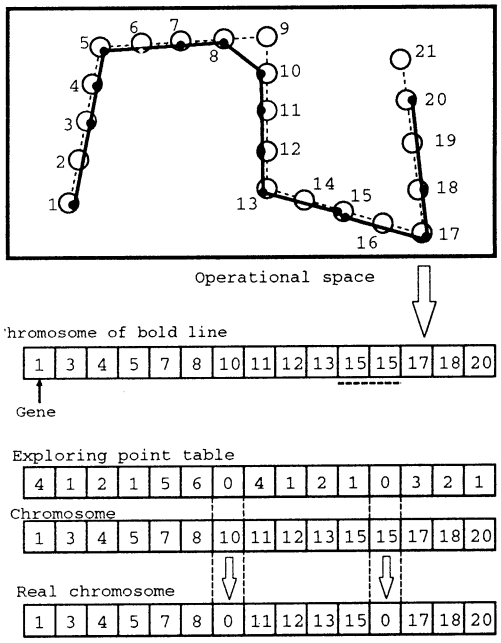


Figure 5: Genetic code for location candidate

Fitness function

The exploring rate defined by the number of explored points/number of points on the operational space for the sequence of operational locations represented by the chromosome is employed here. The path length is the second fitness function when the exploring rate is same as in the operational path.

Genetic operations

Mutation and one-point cross-over is employed here. When no exploring is occurred on the operational locations, an integer '0' is put on that location. Rank selection rule with the elitist strategy is employed for selection of the parents. Then the genetic algorithm is applied as follows:

Step 1: Generate the operational location candidates as chromosomes in the population.

Step 2: Evaluate the fitness function for each individual based on the computer simulation.

Step 3: Select the parents based on the rank selection rule with the elitist strategy and produce the offsprings with mutation and crossover. Each chromosome is arranged in order.

Step 4: Repeat Steps 1 to 3 until the stopping condition is satisfied.

4 Computer simulation results

Simulation studies are carried out by using the simple operational space and reference path. We apply the proposed method for the operational space and reference path shown in Fig.6 with the GA–setup listed in Table.1. The modified path at the 60–th generations is shown in Fig.7. The number of operational locations, exploring rate and path length are compared with those of the reference path are shown in Table2.

Table 1 GA setup used in simulation

	GA for path modification	GA for operational location selection
population size	50	180
mutation rate	0.02	0.02
crossover rate	1.0	1.0
generations	60	70

Table 2 The result of simulation

	number of operational locations	exploring rate	path length
reference path	161	395/400	2250.0
modified path	127	400/400	1216.8

Table 3 The result of simulation(2)

	number of operational locations	exploring rate	path length
reference path	97	199/214	691.3
modified path	78	214/214	637.9

Then we apply this method for another operational space and reference path shown in Fig.8 and the number of operational locations, exploring rate and path length are compared with those of the reference path are shown in Table3. Since these results show that the proposed method provides a shorter path to explore all the points, this approach has the validity to path planning problem for the robot considered in this paper.

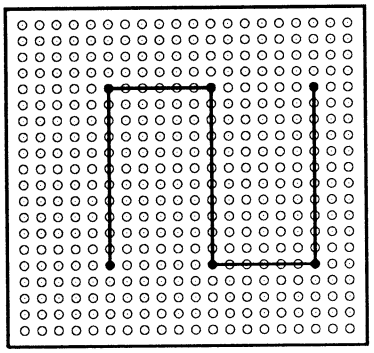


Figure 6: Reference path

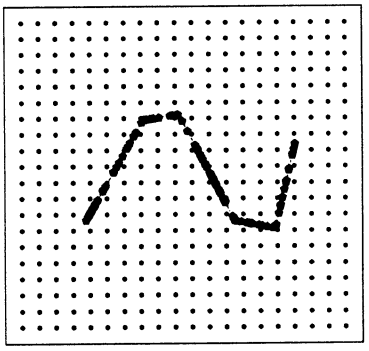


Figure 7: The modified path by proposed method

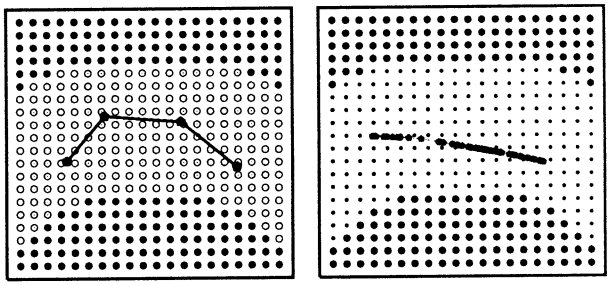


Figure 8: Reference path and the modified path by proposed method

5 Conclusions

In this paper, we have considered a path planning procedure for the auto-mobile robot with four independent arms and presented the improvement procedure of the reference path by using the genetic algorithms. Genetic algorithms are used to improve the reference path and determine the suitable operational locations. Some numerical simulation results have illustrated the validity of this approach. It is further issue to apply the actual operational space.

References

[1] C.-K.Yap, "Algorithmic motion planning," *in Advances in Robotics, Vol.1: Algorithmic and Geometric Aspects of Robotics*, J.T. Schwarts and C.-K. Yap, Eds. Hillsdale, NJ:Lawrence Erlbaum, 1987, pp.95–143.

[2] Jing Xiao, Zbigniew Michalewicz, Lixin Zhang and Krzystof Trojanowski, "Adaptive Evolutionay Planner/Navigator for Mobile Robot ," *IEEE trans. of Evolutionary Computation*, Vol.1, No.1, pp.18–28, 1998.

[3] Takuno. M.et.al.:Path planning algorithm for auto-mobile examiner robot, *Preprints of 6th SICE Chugoku branch annual conference*, pp.54–55(1997).(In Japanese)

[4] Goldberg, G. A., *Genetic Algorithms*, Addison-Wesley, Massachusetts (1989).

[5] Bäck, T. *Evolutionary Computation*, Oxford Press, New York, (1996).

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

An Evolvable NAND-Logic Circuit Applied to a Real Mobile Robot Khepera

Yu Wei, Md. Monirul Islam, Ryoichi Odagiri, Tatsuya Asai and Kazuyuki Murase
Department of Information Science, Fukui University,
3-9-1 Bunkyo, Fukui 910-8507, Japan

Abstract

A logic-circuit controller was proposed here as a realistic and minimum model of an evolvable hardware, and the validity of the approach was tested with a simulator loaded on a real mobile robot Khepera. The controller consisted of 8 or 12 NAND gates, and 4 of which drove the motors. Sensor outputs were coded in an all-or-none fashion. A matrix representing the connections of the NAND gates was evolved in an environment with obstacles. Khepera successfully evolved to navigate in the environment. The best evolved controller contained sequential logics, which could generate a series of complex behaviors. The genetically developed best individuals were compared wth a neural network controlled robot whose coefficients had been adjusted by trail and error for good navigation in the environment.In conclusion, programmable ICs of the fused-array type, which were simpler in the structure, would be sufficient for the controller of evolutionary robots. Such a controller could easily be applied to an evolvable hardware, such as a Field Programmable Gate Array.

1 Introduction

Any logic circuit can be constructed from several NAND elements only because they consist as a complete set. In this study, we tried to realize an evolvable hardware simulator — a NAND logic circuit system which controls an autonomous robot. We described the elementary robot behaviors by simplest logics which may also be called reflexes. The information is stored in a distributed manner and behavior is generated by combinations of simple reflexes[1]. By genetically evolving the logic circuit, autonomous behaviors such as navigation and obstacle avoidance were achieved in an experimental environment.

In the first part this paper presents the method of this study: an all-NAND logic circuit simulator implemented in the control micro processor of a mobile robot with introduction of an experimental environment and its task. In the second part, it describes the reactive behavior control system that was based on NAND logic circuit controller. In the third part, it presents the evolution mechanism applied and the related series of experiments conducted to find the appropriate circuit map that could generate the desired autonomous behavior. Finally, we discuss the experimental results and make an analysis on the genetic parameters adopted in the series of experiments.

2 Experimental Environment & Task

The robot we used is a miniature mobile robot Kephera, which has been widely adopted for AI research. It is 55mm in diameter, 30mm in height, and weighs 70g. It is supported by two motor wheels, each controlled by a DC motor with an incremental encoder. It has 8 infra-red proximity sensors placed around its body. It is connected to a workstation with lightweight cables and rotating contacts.

The micro processor (MC68331, 16MHz) installed on the Khepara robot can execute programs written by the user and can be downloaded from the external computer. Multiple processes can be executed in parallel by time-sharing

The robot's task is to navigate in a test field by autonomous actions. The experimental field used is shown in Fig.1.

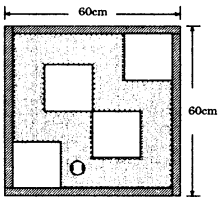


Figure 1: The experimental field

3 The Robot Behavior Control Circuits System

3.1 The behavior control logic circuits

We realized the logic circuits using a single type of logic elements, NAND. All logic elements in the control circuits worked in parallel by time-sharing. The output value of each infra-red proximity sensor was coded to 0 or 1 at a certain threshold We used 8 logic elements in the first series of experiments, and later we expanded the element number to 12.

Each element could have up to 16 input terminals, and the input could be obtained from all the 8 proximity sensors as well as the output(feedback) from all the 8 logic elements. Among the 8 logic elements, outputs from the first 4 elements were assigned to have motor functions; 1st and 2nd ones are to rotate the left motor in a forward and backward direction, respectively, and 3rd and 4th ones to rotate the right motor in a forward and backward direction, respectively. The structure of a 8-NAND logic circuit is shown in Fig.2.

3.2 The Mathematical Expression for the Overall Robot's Behavior

The objective of this evolutionary robot approach is to find a function F of n Boolean variables which represents the desired robot's behavior. Apparently, the domain of the input variable is 2^n, which represents the possible robot-world situations. The input variable is encoded by 8 bit because of the 8 infra-red sensors, and thus represents 256 possible robot-world situations. The associated output of a robot-world situation represents the action it will perform in this situation.

$$Y_k = F_k[(X_0, \ldots, X_{n-1}), (Y_0, \ldots, Y_{N-1})], k = 0, \ldots, N-1$$

where $X_i \in \{0, 1\}$ codes the infra-red sensors, and Y_i represents the function for a single logic NAND element i of the circuits. The overall behavior is defined by Y which is composed of all the functions $(Y_0, \ldots, Y_{N-1})$ for every NAND logic element in the control system.

4 Evolvable Behavior Control System

4.1 The Genetic Information Encoding

While the phenotype was the logic control circuit, the genotype was defined by a sequence of Boolean numbers. A logic element of the 8-NAND circuit was coded with 16 Boolean numbers as shown in Fig.3. The initial 8 numbers represented the connections from 8 proximity sensors, and the last 8 numbers from outputs of all the 8-NAND logic elements. Thus the volume of the connection matrix for the 8-NAND circuit was $8 \times (8 + 8) = 128$ while for the 12-NAND circuit was $12 \times (12 + 8) = 240$.

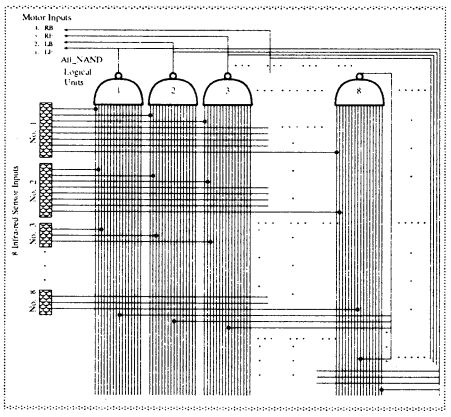


Figure 2: The Structure of a 8-NAND Logic Circuit System

4.2 Performance Fitness Evalutation

The fitness function was designed to select individuals which (1) moved a long distance,(2) as straight as possible, (3) without getting close to any obstacles. That is, the behavior is basically evaluated by the following function,

$$E = D \times (1 - V) \times (1 - S)$$

where D,V and S represent the total mileage of both left and right motors, the difference of mileage between the two motors, and the total of all the sensor outputs. Some additional rules such as thresholds and limits were also used. The state of the robot was observed at every 100ms by itself. Individuals were allowed to move in the test field for a certain period of time, and each individual was evaluated by the total sum of Es obtained at every 100ms during this period.

5 Experiments & Results

During the experiments, the robot was allowed to move 3 sec to change the starting point from the final

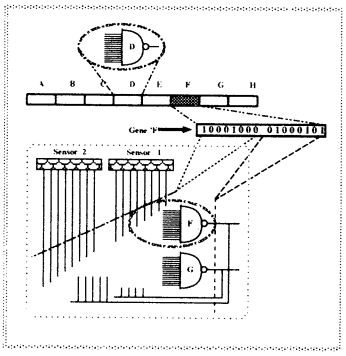


Figure 3: The Sample Image of Encoding for a Single Element 'F'

place of the last trail, and the robot had the identical mechanism to the Braitenberg 's vehicle for this task. The robot then performed its navigation task for 7 seconds, 12 seconds, or 20 seconds.

Examples of the fitness curves are shown in Fig.4 and Fig.5 for the 8-NAND and 12-NAND logic circuits, respectively. The curves with lighter color are fitness evaluation curves obtained by applying the neural network(Braitenberg algorithm) theory in the same experiment. From these figures, we can see the fitness evaluations increased hierarchically. The genotype map of the best individuals evolved in the 8-NAND logic circuit is illustrated in Fig.6. This also shows that the genes changed in a hierarchical way along generations.

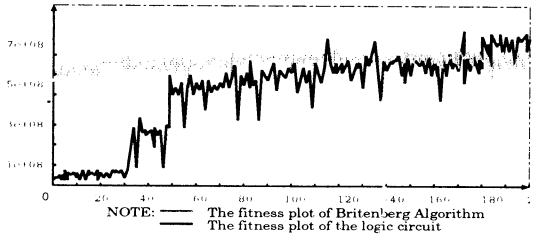


Figure 4: The Fitness Evaluation of the 8-NAND Control Circuit Plotted as a function of generation

Figures 7,8 and 9 show typical behaviors obtained during the evolution of the 8-NAND circuit. The behavior obtained around Generation 70(Fig.7) shows that the robot had learned to avoid obstacles while in its rotating movement. It drived the motor wheel on one side of it with a relatively high speed in the opposite direction to its rotating movement to change its position rapidly. It would keep doing this until the

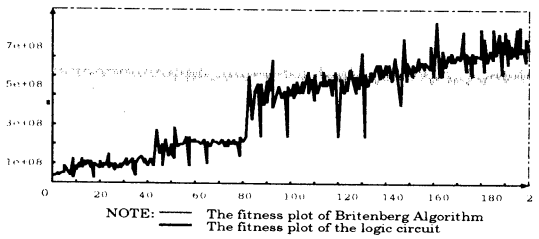


Figure 5: The Fitness Evaluation of the 12-NAND Control Circuit Plotted as a function of generation

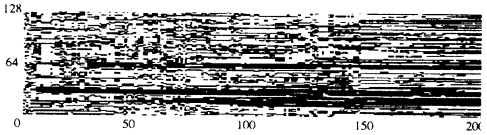


Figure 6: The Genotype Chart for the Best Individual from Generation 0 to 200 (8-NAND)

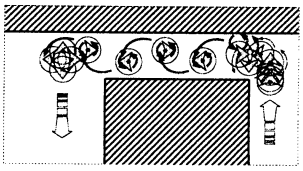


Figure 7: Typical Behavior around Generation 70

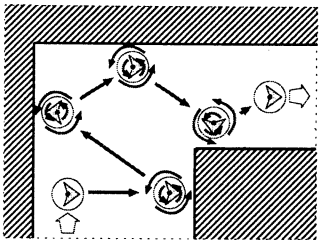


Figure 8: Typical Behavior around Generation 100

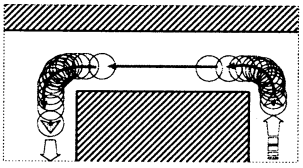


Figure 9: Typical Behavior around Generation 150

robot found a place that was broad enough to perform its continuous rotations.

The behavior obtained around Generation 100 (Fig.8) shows that the robot learned to combine the straight forward motion with the rotate-and-avoid behavior. We can see the fitness value in Fig.4 jumped to a relatively high score. By Generation 150, the robot learned to perform its navigation task in a perfect manner(Fig.9), It achieved a highest fitness score in Fig.4.

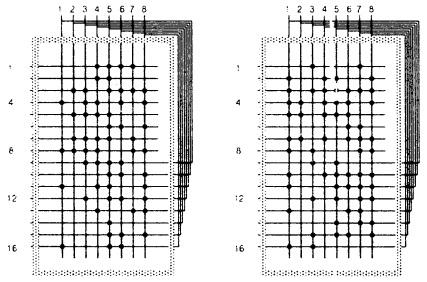


Figure 10: Sample Connection Matrix Images(the best in generation 75 & the best of all generations)

Fig.10 shows a 8-NAND circuit mapping image of the best individual evolved in the entire evolution course. We can see that the control logic circuits were represented by the connection matrix itself: the sensor inputs and feedback inputs go through the NAND gates to generate the outputs to the motion control motors. For comparison, the circuit mapping image of the best individual at Generation 75 is illustrated on the left panel of Fig.10.

We also studied on the relations of the genetic operator and the evolution. We found the total evolution time required to reach the ideal robot performance is almost the same and independent of navigation time used. 7 sec evalution course took 170-220 generations to evolve: the 12 sec evaluation course took 100-130 generations and the 20 sec evaluation took only 60-80 generations to reach the highest behavior performance. We have also found that the best genetic operator combinations are mutation rate of 5%, crossover rate of 90% and elite preservation rate of 30%.

6 Conclusion

In this study, we have presented an evolutionary navigation system of all NAND gate logic control circuit simulator applied on a real mobile robot. The logic circuit control system was able to evolve and learn to navigate and avoid obstacles. Since such a logic circuit controller can easily be applied to an evolvable hardware, these results demonstrated the feasibility of the evolvable hardware to generate robot behavior[2,3,4]. Currently we are trying to extend this method to implement on a real IC platform such as FPGA(Field Programable Gate Array)[5,6].

References

[1] Taku Naito, Ryoichi Odagiri, Yutaka Matsunaga, Manabu Tanifuji, and Kazuyuki Murase. Genetic Evolution of a Logic Circuit Which Controls an Autonomous Mobile Robot(1996). Proceedings of First International Conference, ICES 96, Tsukuba, Japan. Springer-Verlag, October 1996.

[2] Ryoichi Odagiri, WEI Yu, Tatsuya Asai, Osamu Yamakawa, and Kazuyuki Murase(1998). Me asuring the complexity of the real environment with evolutionary robot: Evolution of a real mobile robot Khepera to have a minimal structure. 1998 IEEE world Congress on Compu tational Intelligence, Anchorage, Alaska, U.S.A. May, 1998.

[3] Didier Keymeulen, Marc Durantez, Kenji Konaka, Yasuo Kuniyoshi, and Tetsuya Hi guchi(1996). An evolutionary robot navigation system using a gate-level evolvable hardware. From biology to hardware, ICES96. Tsukuba, Japan. 1996.

[4] Ricardo S. Zebulum, Marco Aurelio Pacheco, and Marley Vellasco(1996). Evolvable Systems in Hardware Design: Taxonomy, Survey and Applications. Proceedings of First International Conference, ICES 96, Tsukuba, Japan. Springer-Verlag, October 1996.

[5] Shinichi Shiratsuchi(1996). FPGA as a Key Component for Reconfigurable System. Proceedings of First International Conference, ICES 96, Tsukuba, Japan. Springer-Verlag, October 1996.

[6] Isamu Kajitani, Tsutomu Hoshino, Daisuke Nishikawa, Hiroshi Yokoi, Shougo Nakaya, Tsukasa Yamauchi, Takeshi Inuo, Nobuki Kajihara, Masaya Iwata, Didier Keymeulen, and Tetsuya Higuchi(1998). A Gate-Level EHW Chip: Implementing GA Operations and Reconfigurable Hardware on a Single LSI. Proceedings of Second International Conference, ICES 98, Lausanne, Switzerland, September 1998. Springer-Verlag, 1998.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Cooperation of Real Mobile Robots Using Communication

Md. Monirul Islam, Yu Wei, Ryoichi Odagiri, T. Asai and K. Murase
Department of Information Science, Fukui University
3-9-1 Bunkyo, Fukui 910-8507, Japan

Abstract

The feasibility of using communication for cooperation between two real mobile robots was investigated and their evolution processes were observed. One of the two miniature mobile robots Khepera was equiped with a vision turret and the other had a marker, a cylinder with black vertical stripes on a white back ground in the front side, on the top of the body. The task given to the robots was to navigate in a field by avoiding obstacles (mainly the wall of the test field) and try to close each other face to face by developing communication between them. To achieve the task, the robot without vision module needs cooperation (e.g. vision data) from other robot by establishing communication with it. In experimnets, at around 150th generation, both robots acquired the capability of obstacle avoidance and to move close each other face to face by developing communication between them.

Keywords- mobile robot, cooperation, communication.

1 Introduction

Cooperation is a form of interaction usually based on communication. Certain types of cooperative behavior depend on directed communication. Specifically, any cooperative behaviors that require negotiation between agents depend on communication in order to assign to particualr task to the participants [1].

In nature most species (e.g. ants, bees) practice exchange information via communication for cooperation and other purposes. For example, bees use signals, such as the waggle dance, with the sole purpose of transmitting information and recruiting. In contrast, they also use cues, such as the direction of their flight, which transmit hive information as a by-product of their other behaviors [2]. Communication in the ants is based on chemical signals. These chemicals are called pheromones and vary from alarm and nesmate rcecogniton. Ants communicates about food resources in a complex three dimensional environment and exhibits a surprisingly sophisticated communicatory system centered around the facultative use of individual-specific trail pheromones [3].

In robotics, compitative coevolution has been a target of many researchers and number of technical articles have been published. But few articles regarding the cooperation among the robots have been reported. We thus attempted to examine the feasibility of cooperation with the two real robots by developing communication between them.

2 Real mobile robot khepera and Vision module

The structure and function of the miniature mobile robot Khepera have been well described elsewhere [4] (Figure 1). In short, it has a diameter of 55 mm and a height of 30 mm, and weighs 70 g. The robot is supported by two wheels, each of which is controlled by a DC motor with an incremental encoder and can rotate in both direction. It has eight infrared proximity sensors placed around its body (six in fornt and two in rear) which are based on emission and reception of infrared light. The proximity sensors have a maximum detection range of 3-5 cm in our environment.

A microprocessor (MC68331, 16MHz) installed in Khepera can execute programs downloaded from external workstation. Sensory information from the infra-red sensors can be acquired and used to control the motors. In addition multiple processes can be executed in parallel by time sharing. Several new single sensors and complete modules (such as vision module, gripper module etc.) can be easily added onto Khepera [5].

In our experiment, One of two miniature mobile robots Khepera was equiped with a vision module K213. We will call this robot the vision-robot. The vision module consists of a CCD-array of 64 photoreceptors. It provides a linear image composed of 64 pixels of 256 gray-levels each, subtending view angle of 36 degree. The optics is designed to bring into fo-

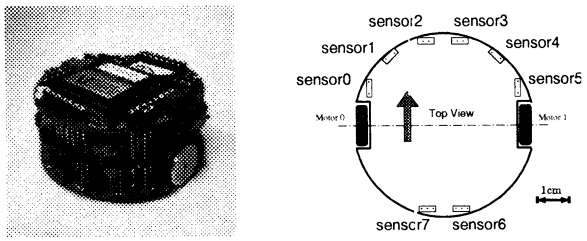


Figure 1: Mobile robot "Khepera"

cous objects at distances between 5cm and 50cm. The K231 module also allows detection of the position in the image corresponding to the pixel with minimal intesity. We exploited this facility by dividing the visual angle into three fileds (left, center, and right).

3 Forms of communication

Communication is the most common means of interaction among intelligent agents. When multiple agents are present in a field, any observable behavior and its consequences can be interpreted as a form of communication. The individuals of insect society practice direct or indirect communication among them [6].

Indirect communication is based on the observed behavior, such as honey bees dance, of other agents. In biological literature, this type of communication is referred to as stigmergic, where it refers to communication based on modification of the enviroment rather than direct message passing [7].

On the other hand, direct communication is a purely communicative act, one with the sole purpose of transmitting information, such as radio communication. Direct communication can be one-to-one or one-to-many, but in both cases the receiver should be identified.

In order to study the role of communication we used one way direct communication, i.e., the vision-robot commuincated its vision information to the other robot via host machine using RS232 serial port. The other robot received the vision information after few milliseconds delay. Therefore, we added the same amonut of delay in the vision-robot to ensure that both robots could start their behaviors at the same time. This limitation could be overcome by using radio communication.

4 Neural network controller

In both robots, simple two-layered neural networks were used to generate motor outputs from the sensory and vision information (Figure 2). The standard genetic algorithm was used to determine the coefficients of the neural network [8]. For maintaining system complexity at a manageable level, the neural network architecure was kept fixed, and only synaptic strenghts were evolved. The network produced two output signals to motors by summing up all the values from eight sensory inputs, three vission inputs and one bias unit with variable coefficients. That is, each output was generated by

$$S_p = S_{default} + G \sum w_i x_i$$

Here, S_p, $S_{default}$, G, w_i and x_i represent the output value to the motor, the base navigation speed of the motor, global gain, connection strength of the sensors and vission, and sensory and vision information repectively. The global gain determines the sensitivity to the modulation signal from sensors and visions. The vaules from sensors were 0-1023 and visions were 0-255. The connection strengths were expressed in 5 digits as described later. The values for $S_{default}$ and G were set to 5 and 1/1600 respectively.

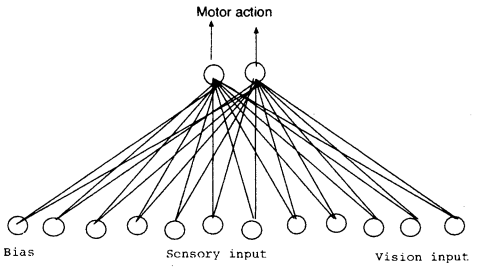


Figure 2: Network architecture

5 Experimental Framework

The two robots were evolved in a square arena of size 47 x 47 cm with high white walls so that the one could always see the other. In our experiments, two robots were attached to a Sun SPARCstation 2 with two serial ports through double serial cable and specially designed rotating contact (Figure 3). To improve refelction of infrared light emitted by the other robot,

the motor bases of the both robots were wrapped by white paper [9]. All low level processes, such as sensor reading, motor control and other monitoring processes were performed by the on-board micro-controller; while other processes (neural network activation and genetic operators) were managed by the Sun CPU. Here the cable allows to keep full track of the robot behavior, power supply to the robot and data communication between robots.

Given the small sizes of the neural controllers under evolution, we used direct genetic encoding [10] on the connection strengths: each connection strength was encoded on five bits, the first bit determining the sign of the synapse and the remaining four bits its strength. Since the visual angle of the vision turret was divided into three fields, the genetype of these robots were 5 x ((8 Sensory cells + 3 Vision cells + 1 Constant cell) x 2 outputs) bits long.

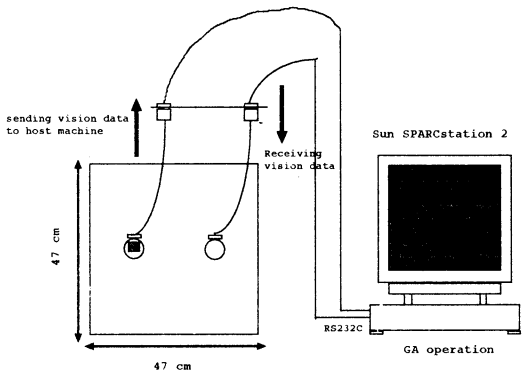


Figure 3: Experimental setup

The input units of the neural networks were attached to the sensors and visions of the robot, and the output unit activations were directly used to set the velocity of the wheels. The robots were left free to move as a result of the activity generated by the neural network while its performances were recorded and accumulated to the pre-design fitness function as described later.

Two separate populations of 10 individuals each were evloved for 200 generations. For each trial, both robots were always positioned on a horizontal line in the middle of the environment at a distance of 15 cm but always at a new random orientation. Each individual was decoded into the corresponding neural network.

Each robot could move for a limited number of actions; each lasting 5 see. There was no synaptic change during the life of each individual. When all the individuals in the population had been tested, three genetic operators -selective reproduction, crossover and mutation- were applied to create completely new population of the same size. The GA parameters used in this experiments were listed in the table I.

TABLE I
THE GA PARAMETERS

Generation	200
Population	10
Gene length	200
Crossover rate	0.7
Mutation Rate	0.02
Elite Rate	0.5

When both robots were closer with face to face, the pixel value of the vission turret and the front sensors value were higher. Therefore, we took the sum of values from the pixels and front sensors as the measure of distance to other robot. The fitness functions, $\phi 1$ for vision-robot which was only the function of one variable, and $\phi 2$ for the other robot which was described as a function of three variables, are given below.

$$\phi 1 = \frac{1}{64}\sum_{i=1}^{64} p(i)/p_{max}$$
$$\phi 2 = (\phi 1 + s/s_{max})/2 * V * \Delta v$$

where p(i) and p_{max} represent the i-th pixel value and the maximum pixel value of the line CCD respectively; s is the average value of the two front sensors; s_{max} is the maximum sensor value; V is the measure of the average rotation speed of the two wheels and Δv is the absolute value of the algebric difference between the sign speed values of the wheels (positve in one direction, negative the ohter). The state of the robot was observed at every 100 ms and the variables of the fitness function were detected at every observation.

6 Results

Both Khepera learned to navigate and try to close each other in less than 200 generations (Figure 4); taking approximately 90 seconds for each generation and 5 hrs for the evalution of 200 generation. However, at around 100th generation the best individuals already exhibited behavior near to the optimal. The interaction between real robot and a physical environment, where some physical laws (such as change in light condition and friction etc.), non-white noise at all level

and various types of malfunctioning (see also [10] [11]) are present, can produce a variety of experimental results. Therefore, we did several runs of the experiment. The results presented here have replicated in several runs.

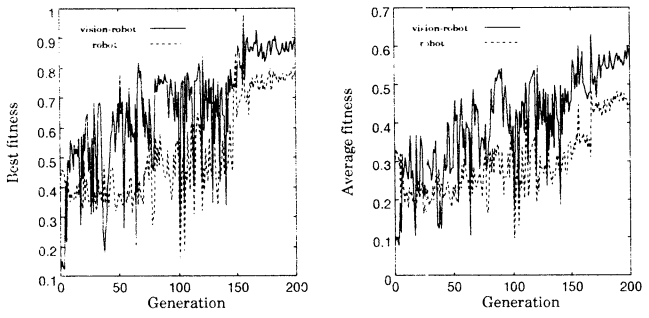


Figure 4: Fitness measured in experiment. Left: Fitness of the best individuals at each generation. Right: Average population fitness

As expected, initially the fitness value of the vision-robot was very low whatever they might did, because the robots were not good to close the other face to face; for the same reason, the other robot score high due to closeness of obstacle. After 150th generations, their navigation was extremely smooth; they never hited the wall and rather they were able to close each other face to face in maintaining a straight trajectory.

7 Conclusions

The main goal of this paper was to show that in muti-agent systems communication among agents can help to develop cooperative behavior. The experimental environment was simple; and the fitness fucntion of each agent was detailed and aimed at developing a specific behavior, i.e., straight navigation with the most appropriate speed and try to close the other. The results displayed the evolution of efficient strategies which partially relied on the autonomous development of the elements of behavior. The simple idea of using communication among the agents is particularly suitable where agents interact based on incomplete sensory information.

References

[1] Maja J Mataric, "Using Communication to Reduce Locality in Distributed Multi-Agent Learning," *Journal of Experimental and Theoretical Artificial Intelligence*, special issue on Learning in DAI Systems, Gerhard Weiss, ed., 10(3), pp. 357-369, Jul-Sep, 1998.

[2] T. D. Seeley, "The honey bee colony as a superorganism," *American Scientist 77*, pp. 12-23, 1989.

[3] J. Klotz, D. Williams, B. Reid, K. Vail and P. G. Koehler, "Ant trails: A key to control with baits," *A Series of the Entomology and Nematology Department, Florida Cooperative Extension Service, Institute of Flood and Agricu lture Sciences, University of Florida*, March 1994.

[4] F. Mondada, E. Franzi and P. Ienne, "Mobile robot miniaturisation: A tool for investigation in control algorithms," *Proceedings of the Thrid International Symposium on Experiment Robotics*, Kyoto, Japan, 1993.

[5] D. Floreano, F. Mondada, "Evolution of homing navigation in a real mobile robot," *IEEE Transactions on Systems, Man and Cybernatics*, Vol. 26, No. 3, pp 396-407, 1996.

[6] Guy Theraulaz, Eric Bonabeau, "Modeling the collective building of complex architecture in social insects with lattice swarms," *Journal of Theoretical Biology*, 177, 381, 1995.

[7] Karl von Frisch, "The Dance Language and Orientation of Bees," *The Belknap Press of Harvard University Press, Cambridge, MA*, 1967.

[8] J. H. Holland, "Adaptation in natural and artificial systems," *The University of Michigan Press*, Ann Arbor, 1977.

[9] D. Floreano, S. Nolfi, F. Mondada, "Competative co-evolutionary robotics: From theory to practice," In R. Pfeifer, *From Animals to Animats IV*, Cambridge, MA: MIT Press, 1998.

[10] X. Yao, "A review of evolutionary artificial neural networks," *International Journal of Intelligent Systems*, Vol. 4, pp. 203-222, 1993.

[11] L. Steels, "Building agents out of autonomous behavior systems," *The "artificial life" to "artificial intelligent" building situated embodid agents*, L. Steels and R. Brook Eds, Lawerence, New Haven, 1993.

[12] L. Steels, "The artificial life robots of artificial intelligence," *The Artificial Life*, Vol. 1, pp. 75-110, 1994.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

SOLVING THE EQUATIONS OF CONSTRAINED MOTION IN A LOWER LIMB MODEL

Cinthia Itiki[1], Robert Kalaba[2] , and Harriet Natsuyama[3]
[1]Universidade de São Paulo, [2]University of Southern California, and [3]Seasons Associates

ABSTRACT: Recently, a new approach to formulating and solving the equations of constrained motion has been developed. The optimization of Gauss' principle of least constraint defines a solution in which the differential equations contain a certain generalized inverse matrix. We have applied this approach to a model of the lower limb in which there are two segments represented by two masses and a foot with a given mass. A muscle connecting the upper and lower segments is represented with viscosity and elasticity parameters. Trajectories of the leg are determined by solving the equations of motion using Runge-Kutta integration. The parameters are varied in a comparative study.

KEY WORDS: constrained motion, generalized inverse method, muscle model, viscosity, elasticity

INTRODUCTION

The purpose of this paper is to investigate the use of generalized inverses in the determination of the trajectory of a simplified lower limb model, which is represented by a constrained mechanical system of particles.

Recently, a new approach [1] to formulating and solving the equations of constrained motion has been developed. The optimization of Gauss' principle of least constraint defines a solution in which the differential equations contain a certain generalized inverse matrix. The acceleration is given explicitly in its generalzed inverse form. The generalized inverse method is equivalent to Lagrange's and Gibbs-Appell's equations of motion [2, 3, 4]. We have applied this approach to obtain the trajectories of a leg model.

Since this work is an exploratory research on the use of the generalized inverse method, it is reasonable to start with a greatly simplified model of the lower limb. The model may be upgraded later, depending on future research interests.

LOWER LIMB MODEL

In this work, the lower limb is represented by two segments, as in figure 1. The first segment represents the thigh, and the second one represents the shank.

The lengths and masses of the limb segments are calculated from tables that give them as percentage of the total body's weight and height [5]. The total body weight, for an average man between 20 and 65 years old, is equal to 78.4 kilograms [6]. His height is equal to 1.755 meters [6]. The estimated lengths of the thigh and shank for this average man are 42.4 and 42.2 centimeters respectively. The center of gravity of the shank is located at 18.3 centimeters from the knee, and the center of gravity of the thigh is at 18.4 centimeters from the hip (which is considered the origin of the coordinate system). The thigh, shank and foot masses are equal to 8.232, 3.724 and 1.121 kilograms respectively.

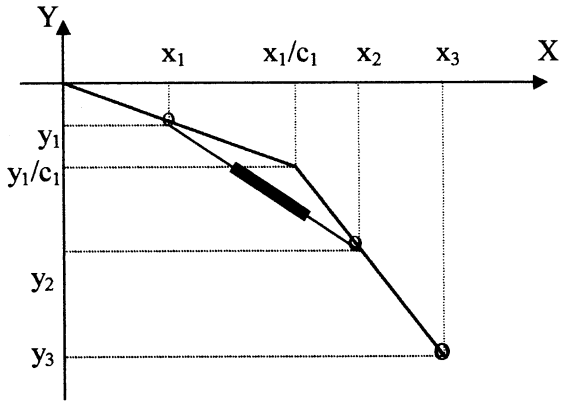


Figure 1 - Lower limb model.

The constraint equations keep the dimensions of the leg segments constant. They are given by

$$x_1^2 + y_1^2 - (c_1 l_1)^2 = 0, \qquad (1)$$

$$\left(x_2 - \frac{x_1}{c_1}\right)^2 + \left(y_2 - \frac{y_1}{c_1}\right)^2 - (c_2 l_2)^2 = 0, \qquad (2)$$

---

[1] Universidade de São Paulo, EPUSP-PEE, Av. Prof. Luciano Gualberto, trav.3, n.158, São Paulo, SP 05508-900, Brasil
The first author thanks FAPESP for the symposium participation grant n. 98/13627-0

$$\left(x_3-\frac{x_1}{c_1}\right)-\frac{1}{c_2}\left(x_2-\frac{x_1}{c_1}\right)=0\,, \tag{3}$$

$$\left(y_3-\frac{y_1}{c_1}\right)-\frac{1}{c_2}\left(y_2-\frac{y_1}{c_1}\right)=0\,, \tag{4}$$

where l_1 and l_2 are the lengths of the thigh and shank. The constants c_1 and c_2 give the ratio between the proximal distances of the centers of mass and the lengths of the leg segments. They are equal to 0.433 and 0.434 respectively.

Two aditional constraints keep the thigh in the horizontal position

$$x_1-c_1\cdot l_1=0\,, \tag{5}$$

$$y_1=0\,. \tag{6}$$

The muscle is represented by a spring in parallel with a dashpot, connecting the shank to the thigh. The direction of the passive force exerted by the muscle is given by the line that connects the centers of mass of the shank (x_2,y_2) and thigh (x_1,y_1). Besides the forces generated by the components of the muscle model, there is also an external force due to gravity g. The total external force is given by

$$\mathbf{f}=\begin{bmatrix}0\\-m_1g\\0\\-m_2g\\0\\-m_3g\end{bmatrix}+\left\{K+B\cdot c(\mathbf{x},\dot{\mathbf{x}})\right\}\begin{bmatrix}(x_2-x_1)\\(y_2-y_1)\\(x_1-x_2)\\(y_1-y_2)\\0\\0\end{bmatrix}, \tag{7}$$

where m_1, m_2 and m_3 are the masses of the thigh, shank and foot; K is the elasticity constant; B is the viscosity constant; g is equal to 9.81 m/s^2; and

$$c(\mathbf{x},\dot{\mathbf{x}})=\frac{(x_2-x_1)(\dot{x}_2-\dot{x}_1)+(y_2-y_1)(\dot{y}_2-\dot{y}_1)}{(x_2-x_1)^2+(y_2-y_1)^2}\,. \tag{8}$$

THE GENERALIZED INVERSE METHOD

In order to apply the generalized inverse method, we need to obtain linear relationships on the acceleration components. The constraint equations are differentiated twice, so that the following relationships are obtained

$$x_1\cdot\ddot{x}_1+y_1\cdot\ddot{y}_1=-\dot{x}_1^2-\dot{y}_1^2\,, \tag{9}$$

$$\begin{aligned}&-\frac{1}{c_1}\left(x_2-\frac{x_1}{c_1}\right)\cdot\ddot{x}_1-\frac{1}{c_1}\left(y_2-\frac{y_1}{c_1}\right)\cdot\ddot{y}_1+\ldots\\&+\left(x_2-\frac{x_1}{c_1}\right)\cdot\ddot{x}_2+\left(y_2-\frac{y_1}{c_1}\right)\cdot\ddot{y}_2=\ldots\\&-\left(\dot{x}_2-\frac{\dot{x}_1}{c_1}\right)^2-\left(\dot{y}_2-\frac{\dot{y}_1}{c_1}\right)^2\end{aligned}\,, \tag{10}$$

$$-\frac{1}{c_1}\left(1-\frac{1}{c_2}\right)\cdot\ddot{x}_1-\frac{1}{c_2}\cdot\ddot{x}_2+\ddot{x}_3=0\,, \tag{11}$$

$$-\frac{1}{c_1}\left(1-\frac{1}{c_2}\right)\cdot\ddot{y}_1-\frac{1}{c_2}\cdot\ddot{y}_2+\ddot{y}_3=0\,, \tag{12}$$

$$\ddot{x}_1=0\,, \tag{13}$$

$$\ddot{y}_1=0\,. \tag{14}$$

We can represent the relationships in matrix notation [1]

$$\mathbf{A}\,\ddot{\mathbf{x}}=\mathbf{b}\,, \tag{15}$$

where $\ddot{\mathbf{x}}$ is the acceleration vector.

The acceleration is obtained in its generalized inverse form [1]

$$\ddot{\mathbf{x}}=\mathbf{M}^{-1}\mathbf{f}+\mathbf{M}^{-1/2}\left(\mathbf{A}\mathbf{M}^{-1/2}\right)^{+}\left(\mathbf{b}-\mathbf{A}\mathbf{M}^{-1}\mathbf{f}\right), \tag{16}$$

where $\mathbf{M}$ is a diagonal matrix containing the elements $\{m_1, m_1, m_2, m_2, m_3, m_3\}$, and $(.)^+$ denotes the generalized inverse matrix of (.) [7].

The trajectories are obtained by numerical integration of the accelerations.

From equations 9 to 15, matrix **A** and vector **b** are determined analytically. From the analytical expressions, MATLAB [8] can calculate the numerical values of matrix **A** and vector **b**. The right hand side of equation 16 can be implemented through dynamic programming [9, 10]. It can also be calculated through the MATLAB built-in function pinv(.). Once the numerical value of the acceleration vector is known, a numerical integration is performed, using a fourth-order Runge-Kutta algorithm. The constant integration step was 0.005 second and the integration time was 2.5 second. The initial conditions were given for a leg in the horizontal position. The initial vertical coordinates were zero, as well as all the initial velocity components. The initial horizontal coordinates of the thigh, shank and foot were given by $(c_1\cdot l_1)$, $(l_1+c_2\cdot l_2)$ and (l_1+l_2) respectively.

RESULTS

Figure 2 shows the obtained results for viscosity values of 180 and 220 N·s/m and elasticity values of 1800 and 2200 N/m.

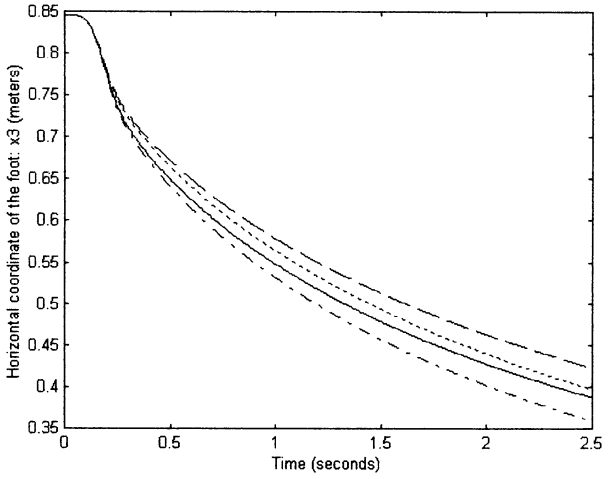


Figure 2 – Horizontal coordinate of the foot:
(——) B=180 N.s/m and K=1800 N/m;
(- - -) B=180 N.s/m and K=2200 N/m;
(- . -) B=220 N.s/m and K=1800 N/m;
(. . .) B=220 N.s/m and K=2200 N/m.

In the continuous line of figure 2, the viscosity and elasticity parameters were 180 N.s/m and 1800 N/m respectively. If there is an increase on the elasticity to 2200 N/m, the horizontal coordinate assumes higher values (dashed line), indicating smaller displacements. If there is only an increase in the viscosity constant, from 180 to 220 N.s/m, then the horizontal coordinate of the foot assumes lower values (dash-dotted line). However, if both elasticity and viscosity constants are increased, the foot has horizontal displacements (dotted line) that are more similar to the initial case (continuous line).

Figure 2 shows that it is not an easy task to determine which combination of parameters generates a given trajectory. This problem is studied in [11].

CONCLUSION

Using a simplified model of the lower limb, we determined the equations of constrained motion through the generalized inverse method. This method allows the use of rectangular coordinates instead of forcing the choice of generalized coordinates. The constraint equations do not need to be independent, as long as they are consistent. Even though we used only holonomic constraints in this work, non-holonomic constraints are even easier to deal with. They are differentiated only once, in order to generate linear relationships on the acceleration components. The inclusion of an additional constraint is easily handled: an extra line appears in matrix **A** and in vector **b**. Another advantage of this method is that the acceleration is given explicitly by equation 16. So, even if the method is not the most efficient one for the specific set of simulations performed in this work, it has a lot of flexibility to handle more complex systems.

The model of the lower limb can be greatly improved by using better models of the muscle [12, 13], and considering the points of origin and insertion of muscles. The bones could also be modeled as rigid bodies. More complex models must be pursued, now that the generalized inverse method showed itself effective in the solution of the equations of constrained motion of a simplified model of the lower limb.

REFERENCES

[1] Udwadia F.E., Kalaba R.E. (1992), A new perspective on constrained motion. *Proc. R. Soc. Lond. A*, v. 439, pp. 407-410

[2] Kalaba R., Udwadia F., Xu R., Itiki C. (1995), The equivalence of Lagrange's equations of motion of the first kind and the generalized inverse form. *Nonlinear World*, n.2, pp.519-526

[3] Itiki C, Kalaba R., Udwadia F. (1995), Appell's equations of motion and the generalized inverse form. In Agarwal R.P. (ed.) *Recent Trends in Optimization Theory and Applications*, World Scientific Series in Applicable Analysis, vol.5, World Scientific, Singapore, pp.123-143

[4] Itiki C. (1996), *Constrained Motion and Generalized Inverses with Applications in Biomechanics*. Ph.D. Dissertation, University of Southern California, Los Angeles

[5] Kreighbaum E., Barthels K.M.(1985), *Biomechanics: a qualitative approach for studying human movement*, 2nd. ed., Burgess, Minneapolis, appendix III, pp. 654-658

[6] Tilley A.R., Henry Dreyfuss Associates. (1993), *The measure of man and woman*, Watson-Guptill, New York, pp. 11

[7] Graybill F.A. (1983), *Matrices with Applications in Statistics*. 2nd. ed., Wadsworth, Pacific Grove, ch. 6, pp. 105-148

[8] *MATLAB Reference Guide* (1992), The MathWorks Inc.

[9] Kalaba R., Natsuyama H., Ueno S. (1998), Regression analysis via dynamic programming: I. Theory. *ISCIE SSS Conference*, Kyoto, November 1998

[10] Kalaba R., Natsuyama H., Ueno S. (1998), Regression analysis via dynamic programming: II. Computational Results. *ISCIE SSS Conference*, Kyoto, November 1998.

[11] Itiki C., Natsuyama H., Kalaba R. (1999), Estimation of muscle parameters of a lower limb model. In: Sugisaka, M. (ed.), *Proceedings of the International Symposium on Artificial Life and Robotics (AROB 4th)*, Beppu, Oita, Japan, Jan. 19-22, 1999

[12] Winter D.A. (1990), *Biomechanics and Motor Control of Human Movement*. 2nd. ed., John Wiley, New York, ch.7, pp.165-189

[13] Zajac F.E. (1989), Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control. *Critical Reviews in Biomedical Engineering*, v.17, n.4, pp.359-411

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

ESTIMATION OF MUSCLE PARAMETERS OF A LOWER LIMB MODEL

Cinthia Itiki[1], Robert Kalaba[2] , and Harriet Natsuyama[3]
[1]Universidade de São Paulo, [2]University of Southern California, and [3]Seasons Associates

ABSTRACT: The purpose of this work is to investigate the use of associative memories in the solution of an inverse problem: the estimation of the viscosity and elasticity constants of a lower limb model. Associative memories have been successful in providing good parameter estimates in a wide range of applications. In this method, the associative memory matrix is first trained using some known sets of stimuli (trajectories of the leg) and responses (muscle parameters). The interaction between this memory matrix and observations of the trajectory of the leg provides estimates of the muscle parameters. Using only four training cases, we obtained estimates with errors as low as two percent.

KEY WORDS: viscosity, elasticity, parameter estimation, associative memories

INTRODUCTION

Associative memories [1] have been successful in providing good parameter estimates in a wide range of applications, including urban planning [2], economics [3] and transport problems [4].

The purpose of this work is to investigate the use of associative memories into the estimation of elasticity and viscosity parameters of a muscle model.

Information about a system behavior may be stored in a memory. This memory is generated by associating the system responses to training stimuli, as seen in figure 1.

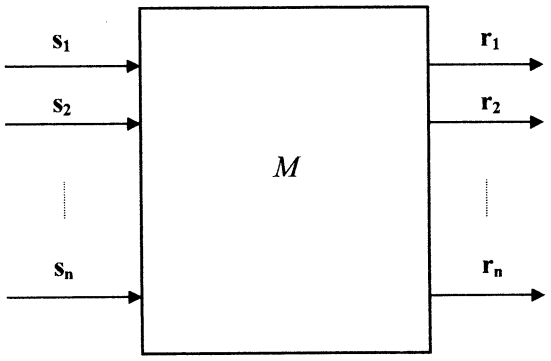


Figure 1 - Associative memories

In this work, we may apply the concept of associative memory by considering the trajectory of a leg as a stimulus vector; and forces or unknown parameters as a response vector to be estimated. The lower limb used in this work is described in [5]. It consists of three masses representing thigh, shank and foot. The thigh and shank centers of mass are connected by a muscle which is represented by a dashpot in paralell to a spring.

RESPONSES

Each response is represented by a vector of size p. The size p of the response vector is the number of muscle parameters to be estimated. The response vector to the i-th stimulus is given by

$$\mathbf{r}_i = \begin{bmatrix} r_i(1) \\ r_i(2) \\ t_i(3) \\ \vdots \\ r_i(p) \end{bmatrix}; \qquad (1)$$

for $i = 1, 2, ..., n$. The number n of response vectors is the number of training cases.

The responses may be concatenated in a response matrix R, so that each response is represented by one of the columns of the matrix

$$R = \left[\mathbf{r}_1 \quad \mathbf{r}_2 \quad \mathbf{r}_3 \cdots \mathbf{r}_n\right]. \qquad (2)$$

In this work, the dashpot viscosity (B) and the spring elasticity (K) constants are the unknown parameters to be estimated. The chosen training case values for viscosity and elasticity are given by

$$R = \begin{bmatrix} 180 & 180 & 220 & 220 \\ 1800 & 2200 & 1800 & 2200 \end{bmatrix}. \qquad (3)$$

STIMULI

Each stimulus is represented by a vector of size m. The size m of the stimulus vector is the number of trajectory samples in time. The i-th stimulus is given by

---

[1] Universidade de São Paulo, EPUSP-PEE, Av. Prof. Luciano Gualberto, trav.3, n.158, São Paulo, SP 05508-900, Brasil
The first author thanks FAPESP for the symposium participation grant n. 98/13627-0

$$\mathbf{s_i} = \begin{bmatrix} s_i(1) \\ s_i(2) \\ s_i(3) \\ \vdots \\ s_i(m) \end{bmatrix}; \quad (4)$$

for $i = 1, 2, ..., n$.

The stimulus matrix S is built so that the i-th column represents the i-th stimulus

$$S = [\mathbf{s_1} \quad \mathbf{s_2} \quad \mathbf{s_3} \cdots \mathbf{s_n}]. \quad (5)$$

In this work, the stimulus matrix is given by the horizontal coordinate x_3 of the foot. The equations of motion for the lower limb model were obtained and integrated for the training case values of viscosity and elasticity, as described in [5]. The obtained horizontal trajectories of the foot were sampled at the rate of four samples per second. In this way, the number m of samples per stimulus was reduced from 500 to 11. The remaining samples were organized in the stimulus matrix

$$S = \begin{bmatrix} 0.8460 & 0.8460 & 0.8460 & 0.8460 \\ 0.7367 & 0.7477 & 0.7336 & 0.7450 \\ 0.6495 & 0.6719 & 0.6408 & 0.6644 \\ 0.5927 & 0.6201 & 0.5796 & 0.6087 \\ 0.5476 & 0.5782 & 0.5310 & 0.5637 \\ 0.5102 & 0.5430 & 0.4908 & 0.5258 \\ 0.4785 & 0.5127 & 0.4567 & 0.4932 \\ 0.4511 & 0.4861 & 0.4273 & 0.4647 \\ 0.4272 & 0.4627 & 0.4018 & 0.4395 \\ 0.4062 & 0.4417 & 0.3793 & 0.4171 \\ 0.3875 & 0.4229 & 0.3594 & 0.3970 \end{bmatrix}. \quad (6)$$

ASSOCIATIVE MEMORY

An associative memory is formed by previous experience acquired through training. The memory matrix is such that the sum of the squares of the elements in the matrix $(R - M \cdot S)$ is minimized. The memory matrix is calculated by

$$M = R \cdot S^{+}, \quad (7)$$

as Kohonen [1] pointed out. The matrix S^{+} is the Moore-Penrose inverse of the matrix S.

PARAMETER ESTIMATES

Once a memory matrix is formed, a prediction of the response may be obtained by using the measured observations **s**. The response estimate is given by

$$\hat{\mathbf{r}} = M \cdot \mathbf{s}. \quad (8)$$

In this work, the observations are generated synthetically, in order to validate the method. The horizontal coordinates are obtained for an elasticity of 2000 N/m and a viscosity of 200 N·s/m. The observation vector **s** is given by

$$\mathbf{s} = \begin{bmatrix} 0.8460 \\ 0.7411 \\ 0.6575 \\ 0.6013 \\ 0.5562 \\ 0.5185 \\ 0.4863 \\ 0.4582 \\ 0.4336 \\ 0.4118 \end{bmatrix}. \quad (9)$$

The calculated estimates of elasticity and viscosity are $\hat{K} = 2015.8$ N/m and $\hat{B} = 202.0$ N·s/m which are very good estimates of the true values. The estimation errors are less than 2%.

CONCLUSION

As shown in this work, associative memories may be used in the estimation of muscle model parameters. Two parameters of a muscle model were estimated with less than 2% of error.

The choice of training cases, including number of cases and parameter values is an important and difficult task. In this work, the observations were generated synthetically by computer. As a consequence, the true values were already known. Because of that, the training cases were chosen in the vicinity of the known true values. However, in a real identification problem, we may not have a good idea of where to locate the training cases. It may happen that as the true value of the parameter is out of the range of the chosen training cases, we may find serious limitations. However, alternatives may be sought.

We may choose a new set of training cases around the first estimate, so that a more precise value may be found. There is also the possibility of using a recursive method to build the memory matrix [2]. If the difference between stimuli vectors get smaller, errors may increase. One solution would be the use of multicriteria associative memories [3], another would be the use of training cases with noise [3]. The choice of each one of the above approaches depends on the specific problem to be studied.

It would be highly desirable to obtain biomechanical measurements and choose the appropriate approach for the observed data, instead of applying the different approaches to computer-generated observations.

In order to do so, more realistic models of lower limbs and muscles should be used, along with the observations obtained through biomechanical measurements.

This work shows that associative memories have the potential of being used in the estimation of muscle parameters. So, we conclude that all the above sugestions should be pursued.

REFERENCES

[1] Kohonen T. (1988), *Self-Organization and Adaptive Memory*, Springer-Verlag, New York

[2] Xu R. (1995), *Calibration and Sensitivity Analysis of Nonlinear Systems with Application to Gravity Models*, ch. 3, pp. 25-64, Ph.D. Dissertation, School of Urban and Regional Planning, University of Southern California

[3] Kalaba R., Tesfatsion L. (1991) Obtaining initial parameter estimates for nonlinear systems using multicriteria associative memories. *Computer Science in Economics and Management*, v. 4, pp. 237-259.

[4] Kalaba R.E., Natsuyama H.H., Ueno S. (1998), The estimation of parameters in time-dependent transport problems: dynamic programming and associative memories. To appear in *Computers and Mathematics with Applications.*

[5] Itiki C., Natsuyama H., Kalaba R. (1999), Solving the equations of constrained motion in a lower limb model. In: Sugisaka, M. (ed.), *Proceedings of the International Symposium on Artificial Life and Robotics (AROB 4th)*, Beppu, Oita, Japan, Jan. 19-22, 1999

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A principle of design of an autonomous mobile robot

Kazuo Tsuchiya, and Katsuyoshi Tsujita
Dept. of Aeronautics and Astronautics, Kyoto University, Kyoto 606-8501, Japan

Abstract

In this paper, it is proposed that a principle of design of an autonomous mobile robot, i.e., a walking robot carrying out a task in the real world from dynamic systems perspective. The control system is composed of the modules, each of which has its own function, cognition, planning and motion control. The modules are realized in terms of a dual dynamic system. A dual dynamic system is composed of a rule dynamics and an element dynamics; An element dynamics processes input signals under the control of the rule dynamics. The outputs of the modules are combined and realize a task specific motion of the robot.

1 Introduction

Research on a walking robot is proceeding actively[1]. A walking robot is a mechanical system with legs composed of links. The control of a walking robot is to make the robot walk stably and efficiently to a target. Currently, the study on the control of a walking robot is done under the condition that a desired motion of the robot is given. At that time, the difficulty of the control of a walking robot is to control the motions of a lot of elements with nonlinear interactions. The motion control of a walking robot has been achieved by a model based control[2]; The inverse kinematics and the inverse dynamics of the robot are preprogrammed and when the desired motion is given, the motion of each link is controlled on the basis of the inverse models. In the future, it is required that a walking robot which can carry out tasks in the real world, where the geometric and kinematic conditions of the environment are not specially structured. In such an environment, first, the robot has to extract the parameters characterizing the environment from sensor signals and compose a desired motion carrying out the tasks and then, control motions of the links realizing the desired motion. The difficulty of the control of a walking robot carrying out a task in the real world is not only the difficulty to control the motions of a lot of elements with nonlinear interactions, but also the difficulty to form a task specific pattern of the motion of a lot of elements.

Now, mechanisms of motion of animals have been studied in ethology[3]. From the view point of behaviorism, the motion of an animal has been modeled as a series of actions caused by the environment. Then, by ethologists, each one of the actions has been revealed to be developed from a set of simple, stereotyped movements of muscle activities. A body of an animal is composed of a lot of joints and muscles. During a motion, a lot of elements are organized into a collective unit to be controlled as if it had fewer degrees of freedom and yet to retain the necessary flexibility for changing internal and external contexts. The motion of an animal seems to offer solutions to the problem to control a lot of elements and the problem to form a task specific pattern[4]. Recently, a motion of an animal have been studied from the dynamic systems perspective. The dynamic model of the behavior of a single celled animal has been proposed by Oosawa[5]; A single cell is modeled as a particle with internal variables where the state of the animal is determined by internal variables and the internal variables vary according to the input signals. The action of the animal to the input signals is determined not by the input signals directly but by the state of the animal and then, the motion of the animal realizes a context-conditioned variability. Kelso[6] has investigated the motion of an animal from the viewpoint of Synergetics; The motion of an animal results from the processes of self-organization and a task specific motion appears when a certain control parameter is scaled to some critical magnitude.

On the other hand, based on the latest achievements of neurobiology and ethology, a new approach to robotics has been developed. Brooks[7] has proposed the subsumption architecture as principle of design of an autonomous mobile robot which can carry out a task in the real world; The control system is composed of behavior-generating units. Each of units responds to the changes in the environment and generates a stereotyped action. Responses from all the units compensate each other and one of which determines the action of the robot. Using the subsumption architecture, Brooks[8] developed a six-legged robot, Genghis to walk over a rough terrain. Through the trajectory of the body is not specified, the robot successfully navigates a rough terrain. When the geometric and kinematic conditions of the environment are complicated, it is required that the motions of the robot become to be complex and a lot of behavior-generating units are needed. At that time, a mechanism to manage a lot

of units may be necessary. In this paper, we will propose a principle of design of a walking robot carrying out a task in the real world from a dynamic systems perspective.

2 Design Principle

In this chapter, a design principle of control system of a walking robot is proposed. In order to make the issues clear, a quadruped locomotion robot is picked up as an example of this kind of robot (Fig. 2.1).

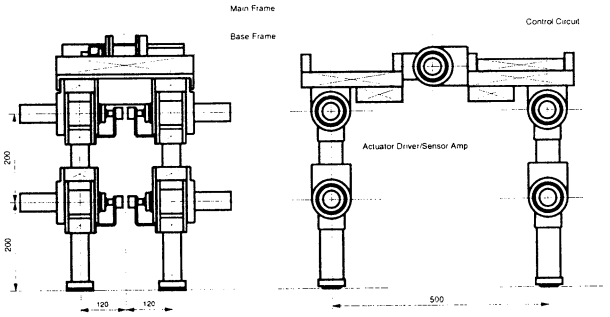


Fig. 2.1 Quadruped locomotion machine

This model has two joints on each leg and two joints on the main body. At each joint, motors are equipped as actuators. This model has three kinds of sensor; visual sensors, reaction force sensors equipped at the end of legs and angle sensors at the motors. The task to the robot is to walk safely and efficiently to a target; The role of the control system is to recognize the environments by the sensor signals and to generate a gait pattern adapted to the environments and to output the control command to the motors which realize the gait pattern. That is to say, the function of the control system is the optimization, i.e., to generate the optimal gait pattern on real-time under the geometric and dynamic constraints of the environments. In order to design a control system as an effective optimization system, three principles are adopted. The first principle is to divide the optimization problem considered into some small problems. In this case, it is divided into following three problems (modules); cognition module, planning module and motion control module. The cognition module derives the model of the environments by using the input signals, i.e., to extract the parameters characterizing the environments; The shape and the inclinations of the paths are estimated by using the input signal of the visual sensor and dynamic state of the system is estimated by using the input signal of the reaction force sensors and the angle sensors. The planning module constructs a desired pattern of motion from the given task, that is, to generate a gait pattern of the robot. The motion control module calculate the control command to the actuators from the desired motion. The control command is composed of the feed-forward command based on the desired pattern of motion and the feedback command based on the sensor signals. The processes in these modules are not necessarily executed successively. The second principle is the inner model principle. That is, each optimization module has its own inner model. The inner model is used to construct or to restrict the search space of the optimization. In the cognition module, the models of the passageways are derived by using the input signals of the visual sensors. This derivation is executed based on the Bayes estimation using a priori informations of the environment, where a priori informations of the environment play a role of the inner model. On the other hand, in the motion control module the control command to the actuators are calculated by using the desired motion. This calculation is established by using the inverse kinematics model and the inverse dynamics model of the system where these inverse models play roles of the inner models. These inner models are not static. They will change slowly through the comparison of the results of the behavior with the given tasks. The last principle is that of distributed parallel processing. The optimization processes are considered as to select or to construct the best assumption in a set of the elements of assumptions under some constraints. In order to make it in terms of the distributed parallel processing, following strategies are adopted; The elemental assumptions are coded into the dynamic elements which has own dynamics. Then, the interactions between the elements are embedded corresponding to the constraints(inner model). The dynamic system constructed in this way can construct the best assumption through their own dynamics as the saptio-temporal pattern on the elements. For example, the model of the environment is derived in the cognition module from the signal of CCD camera. The dynamic elements of the cognition module are assigned to the pixels of the CCD camera and the interactions between the elements are embedded according to the inner model as the priori informations. When the outputs of the CCD cameras are input to the elements, the elements extract the parameters of the environments through their dynamics. Figure 2.2 shows the control system proposed.

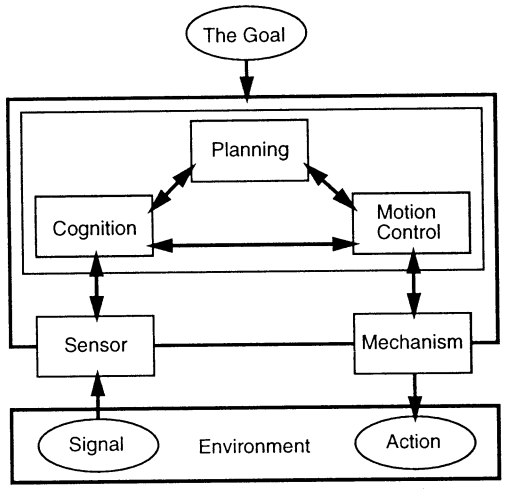


Fig. 2.2 Control system

3 Design method

The control system proposed is composed of modules and each module is the dynamic system composed of dynamic elements. This dynamic system is two time scaled system with the fast variables and the slow ones. The slow variables express the inner model and change slowly through the evaluation of the behavior from the given tasks. The fast variables deal with the external inputs or the inputs from the other modules under the control of the slow variables. The spatio-temporal pattern emerges on the fast variables. The typical example of this class of dynamic system is an artificial neural network model with plastic synapses. Here, a design method of each module proposed based on the dual dynamic system is mentioned[9]. The dual dynamic system is composed of the fast dynamics (element dynamics) of the fast variables and the slow dynamics (rule dynamics) of the slow variables. The fast variables change their values under the constraints of the slow variables. The constraint to the first dynamics is given as a positive function of the fast variables u_i as $H^{(m)}(u_i,\ v_j)$, $m = 1, \cdots, M$, where v_j are the external inputs or the inputs from the other modules. The outputs of the fast dynamics are the spatio-temporal patterns emerged on variables u_i. Here, suppose that the required outputs are static spatial patterns. It is required that each module has the various spatial patterns as possible under the given constraints in order to make the system adaptive to the environments. This means that when we design the first dynamic system with a stochastic differential equation, it makes various equilibrium solutions as possible under the given constraint $H^{(m)}(u_i,\ v_j)$ $m = 1, \cdots, M$. ,that is, the absolute entropy $S = \int p(u_i|v_j) \ln p(u_i|v_j) du_i$ defined with probability density $p(u_i|v_j)$ at the equilibrium states becomes maximum. According to the maximal entropy principle, probability density $p(u_i|v_j)$ is expressed by using constraint $H^{(m)}(u_i,\ v_j)$ as follows;

$$p(u_i|v_j) = Z^{-1} \exp\left(-\sum \lambda_m H^{(m)}(u_i,\ v_j)\right) \qquad (1)$$

where, Z is a partition function and λ_m is Lagrangian multiplier corresponding to constraint $H^{(m)}(u_i,\ v_j)$. Lagrangian multipliers λ_m are adopted as the slow variables in the dual dynamic system. The stochastic differential equation which realizes probability density (1) is given as a Langevin equation which satisfies the detail balance conditions. One of the simple forms is given as follows;

$$\dot{u}_i = -\sum_m \lambda_m \frac{\partial H^{(m)}(u_j,\ v_l)}{\partial u_i} + w_i \qquad (i = 1, \cdots, N) \qquad (2)$$

where, w_i is the white Gaussian noise. The performances of the system are evaluated based on the given tasks. Here, suppose that the performances are evaluated in each module and also suppose that the slow variables λ_m have effects on the performances of the modules through function $g^{(m)}(u_i)$. The desired value of function $g^{(m)}(u_i)$ is given as probability density $\hat{p}^{(m)}\left[g^{(m)}(u_i)\right]$. By using probability density $p^{(m)}\left[g^{(m)}(u_i)|\lambda_m\right]$ calculated from Eq.(refeq:eq2). Relative entropy $K^{(m)}(\lambda_m)$ of probability density $\hat{p}^{(m)}\left[g^{(m)}(u_i)\right]$ in terms of $p^{(m)}\left[g^{(m)}(u)_i|\lambda_m\right]$ is defined as

$$K^{(m)}(\lambda_m) = \int \hat{p}^{(m)}\left[g^{(m)}(u_i)\right] \times \ln\left\{\hat{p}^{(m)}(g^{(m)}(u_i))/p^{(m)}\left[g^{(m)}(u_i)|\lambda_m\right]\right\}$$

The slow variable λ_m is designed so that relative entropy $K^{(m)}(\lambda_m)$ decreases.

$$\dot{\lambda}_m = -\frac{1}{\tau_m}\frac{\partial K^{(m)}(\lambda_m)}{\partial \lambda_m} \qquad (3)$$

where, τ_m is designed so that the dynamics of variable λ_m is slow enough compared with that of variable u_i. When each module is designed based on the dual dynamic system, each module is composed of the two dynamic systems; One is the element dynamics and the other is the rule dynamics. These dynamic systems have complimentary properties; In the element dynamics, the absolute entropy increases with time while in the rule dynamics, the relative entropy decreases with time ,that is, the former is a Markov process while the latter is a Bayes process. These properties may make the system robust against and adaptive to the changes in the environment.

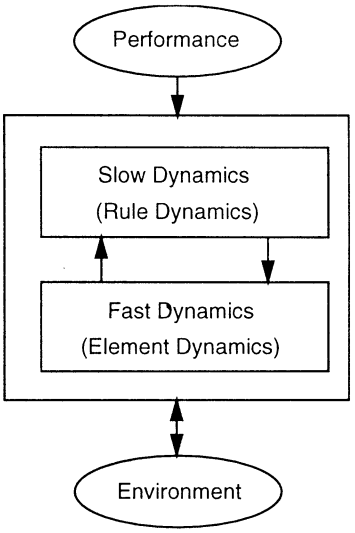


Fig.3.1 Dual dynamic system

4 Summary

This paper has proposed a principle of design of a walking robot carrying out a task in the real world from dynamic systems perspective. The control system proposed is composed of three modules, each of which has its own function, cognition, planning and motion control. The modules are realized in terms of a dual dynamic system. The dual dynamic system is composed of a rule dynamics and an element dynamics where the element dynamics processes the input signals under the control of the rule dynamics. Outputs of the modules determine the task specific behavior of the robot. Hereafter, a hardware mode will be developed based on the principle of design proposed.

References

[1] Int. J. Robotics Research, Vol. 3, No. 2, 1984,
[2] M. Bray, J. Hollerbach, T. Johnson, T. Lozano-Peres, M.Mason Eds., "Robot Motion: Planning and Control," MIT Press, Cambridge, MA, 1982
[3] N. Tinbergen, "The Study of Instinct," Oxford University Press, New York, 1951
[4] N.A.Bernstein, "Coordination and Regulation of Movements," Pergamon Press, New York, 1967
[5] F.Oosawa, "Bio-physics" (in Japanese) Maruzen, Tokyo, 1998
[6] J.A.S.Kelso, "Dynamic Patterns," MIT Press, Cambridge, MA, 1995
[7] R.A.Brooks, "A Robust Layered Control System for a Mobile Robot," IEEE Journal of Robotics and Automation, Vol. 2, No. 1, pp.14/23, 1985
[8] R.A.Brooks, "A Robot that walk; Emergent Behavior from Carefully Evolved Network," Neural computation, Vol. 1, No. 2, pp.253/262, 1989
[9] K. Tsuchiya and K. Tsujita, "Design of a Complex System based on the Maximum Entropy Principle," Artif. Life and Robotics, Vol. 1, pp.65/68

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Autonomous Robot Control by a Neural Network with Dynamic Rearrangement Function

T. Kondo* † A. Ishiguro* Y. Uchikawa* P. Eggenberger**
*Dept. of Computer Science and Engineering
Graduate School of Eng., Nagoya University
Nagoya 464-8603, JAPAN
**Dept. of Computer Science
University of Zurich
Zurich 8057, Switzerland

Abstract

Recently, Evolutionary Robotics approach has been attracting a lot of concerns in the field of robotics and artificial life. In this approach, neural networks are widely used to construct controllers for autonomous mobile agents, since they intrinsically have generalization, noise-tolerant abilities and so on. However, the followings are still open questions; 1) gap between simulated and real environments, 2) evolutionary and learning phase are completely separated, and 3) conflict between stability and evolvability/adaptability. In this article, we try to overcome these problems by incorporating the concept of dynamic rearrangement function of biological neural networks with the use of neuromodulators.

1 Introduction

Recently, Evolutionary Robotics (ER) approach has been attracting a lot of concerns in the field of robotics and artificial life[1, 2]. In contrast to the conventional approaches where designers have to construct robot controllers heuristically, ER approach can construct controllers by taking *embodiment* (i.e. physical size of robot, sensor/motor disposition, etc.) and the *interaction between the robot and its environment* into account. Although ER approach has the above advantages, there still exist the the following drawbacks:

- Whole evolutionary process can not be implemented in the real world due to the time-consuming evolutionary process.
- Due to the above problem, evolutionary processes are usually carried out in the simulated environments. This causes a serious *gap* between simulated and real environments (i.e. evolved agents in the simulated environments are hard to work successfully in the real world).
- Most conventional ER studies are dedicated to *trivial* tasks such as obstacle avoidance, wall following and so on. This implies there still does not exist efficient techniques to evolve agents that can elicit non-trivial tasks.

In ER approaches, artificial neural networks are widely used to construct controllers for autonomous mobile agents[3, 4], since they intrinsically have generalization, non-linear mapping, noise-tolerant abilities and so on.

Another advantage of a neural network-driven robot is that a neural network is a *low-level description of a controller*. It directly maps sensor readings onto motor outputs. Thanks to this, rich emergent properties can be expected through the evolutionary process. Now, question arises. Why are the conventional neural network-driven controllers obtained through the evolutionary process suffering from the above drawbacks? We should notice that conventional neural networks are composed of many uniform threshold units each of which has a non-linear transformation function. Whole information is statically encapsulated in the connecting weights among the neurons. Thus, once the connection weights are determined, the function of the whole neural network is automatically converged to one non-linear mapping. This causes the *process fusion problem* particularly in the case of obtaining non-trivial tasks.

On the other hand, recent neuro-science has revealed that biological neural networks are not static, rather dynamically changing according to the situations. This interesting phenomena (in this article, we call *dynamic rearrangement*) is caused by chemical substances called neuromodulators (hereafter, NMs). In summary, activities of neurons are determined by the current network architecture, and the architecture is also determined by neuron activities. Such a dynamic and mutual coupling yields remarkable adaptation and behavior control mechanisms in the living organisms.

† The order of the authors is arbitrary.

Based on the above considerations, in this study, we apply the concept of dynamic rearrangement of neural networks to ER approaches[6, 7]. We show our approach can yield rich evolvability and adaptability compared to the conventional one by carrying out simulations.

2 Dynamic Rearrangement Function of the Biological Nervous System

The conventional use of the artificial neural networks in the robotics community was that connection weights among neurons are adjusted so that the network can elicit appropriate function according to the desired task. To put it differently, the network properties are encapsulated in the *static* weights.

Recent neuro-science, however, has clarified that neural networks are not static, rather dynamically and quickly changeable. In the followings, we briefly overview a biological evidence of dynamic rearrangement of nervous systems.

A graceful example of the dynamic rearrangement was shown by Meyrand et. al.[5]. They investigated the behavior of lobsters' stomatogastric nervous systems.

As illustrated in Figure 1, this stomatogastric nervous system is mainly consisted of oesophageal, pyloric, and gastric networks. Usually, the individual network shows its own oscillatory behavior. Namely, they work independently. However, once lobsters eat or drink something, these networks are immediately integrated and reconstructed. Consequently, the swallowing network will appear. As in this example, the actual biological neural systems change their structure dynamically according to the situation.

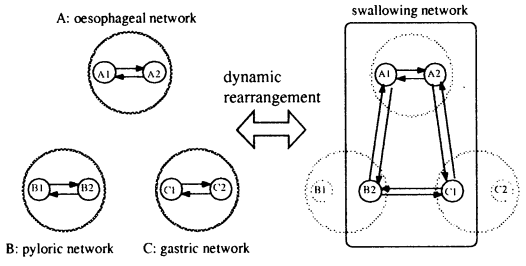


Figure 1: Dynamic rearrangement of a lobster's stomatogastric nervous system[5].

Recent studies in neuro-physiology showed neuromodulators (NMs) play crucial role to regulate this remarkable phenomenon.

3 Proposed Method

3.1 Basic Concept

The basic concept of our proposed method is schematically depicted in Figure 2. As in the figure, unlike the conventional neural networks, we assume that each neuron can potentially diffuse its specific NM according to its activity, and each synapse has receptors for the diffused NMs.

We also assume that each synapse independently interprets the received NMs, and changes its property (e.g. synaptic weight). Through this cyclic interaction between the diffusion and reaction of NMs, we expect that neural networks suitable for the current situation will dynamically emerge (in the figure, the thick and thin lines denote the connections strengthened and weakened by NMs, respectively). Note that this emerged neural network works as a sort of *implicit behavior primitive*.

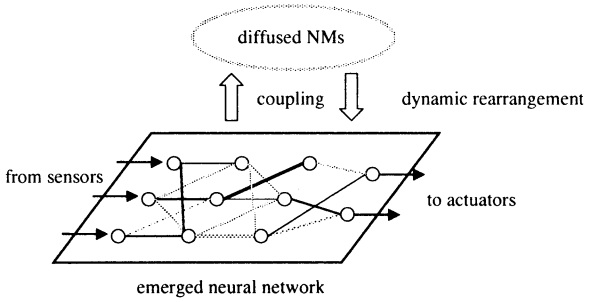


Figure 2: Dynamic rearrangement of neural networks.

In summary, unlike the conventional ER approach that evolves synaptic weights and neuron's bias of neuro-controllers, in this approach we evolve the followings:

- Diffusion of NMs (When, how long, which type of NMs is diffused from each neuron?)
- Reaction to NMs (How does each synapse interpret the received NMs?)

To determine the above parameters, we use Genetic Algorithms (GA). Another advantage to be noted is that even a few number of NMs can yield variety of implicit behavior primitive networks due to *combinatorial effect*.

3.2 Application Problem

In this study, we use a food collecting problem as a practical example. Figure 3 shows this application

problem. Here the task of the robot is to collect the food as many as possible within the prespecified time-step. As in the figure, there are four kinds of environments. Each environment is surrounded by walls, and has food on the lines according to the prespecified distance from the wall listed in Table 1.

Table 1: Location of food.

	Env.1	Env.2	Env.3	Env.4
Distance from Wall	$R \times 2$	$R \times 3$	$R \times 4$	$R \times 5$

$R = radius\ of\ the\ robot's\ body$

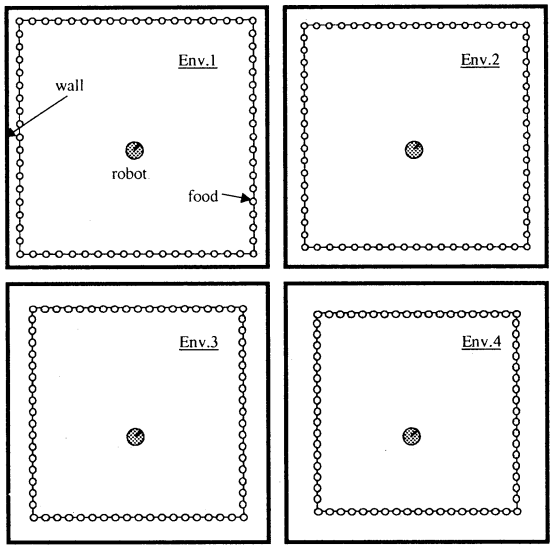


Figure 3: Environments used in the simulations.

Figure 4 illustrates the structure of the robot used in this experiment. This robot is equipped with eight ultrasonic sensors for measuring the distance to the wall, and two wheels driven independently. One infrared sensor is also equipped on its belly to detect the existence of food.

It should be noted that the robot can not know in advance in which environment he will be placed. Therefore, in order to adapt to the encountered environment (i.e. to collect food as many as possible), he has to appropriately correlate his sensor readings and regulate the motor outputs.

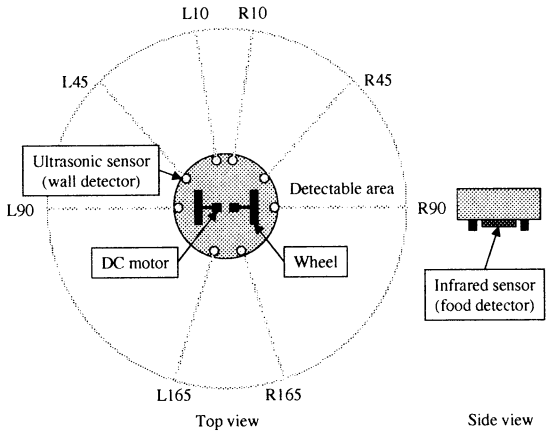


Figure 4: Structure of the robot.

4 Simulation

4.1 Controller

Figure 5 shows the structure of the neural network used in this experiment. As in the figure, this network is consisted of nine sensory neurons, five inter neurons, and two motor neurons. Each sensory neuron just delivers the corresponding sensory reading, and each motor neuron control the velocity of the corresponding wheel.

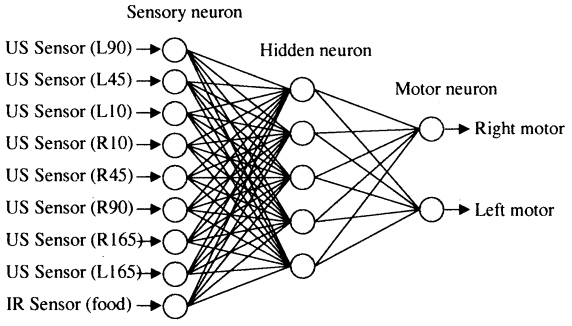


Figure 5: Controller.

The activation of each neuron is determined as:

$$a_i = f(\sum_{j=1}^{N} w_{ji} \cdot a_j - \theta_i) \quad (1)$$

$$f(x) = \frac{1}{1 + e^{-\alpha \cdot x}}, \quad (2)$$

where, a_i represents the activation value of neuron i, w_{ji} indicates the connection weight between neuron i and j. θ_i is the bias input to neuron i. Equation (2) is a squashing function that limits the activation of each neuron between 0.0 and 1.0.

4.2 Proposed NM Model

4.2.1 Diffusion of NMs

As described before, in this model each neuron can diffuse its specific NM and each synapse has receptors for the diffused NMs. Here, for simplicity, we assume that only inter neurons and motor neurons can diffuse NMs, and each of which can diffuse at most one type of NMs.

As in Figure 6, each inter and motor neuron has three parameters to be determined genetically; NM type and upper/lower threshold values. We assume that each neuron can diffuse its own NM while its activation is within the thresholds.

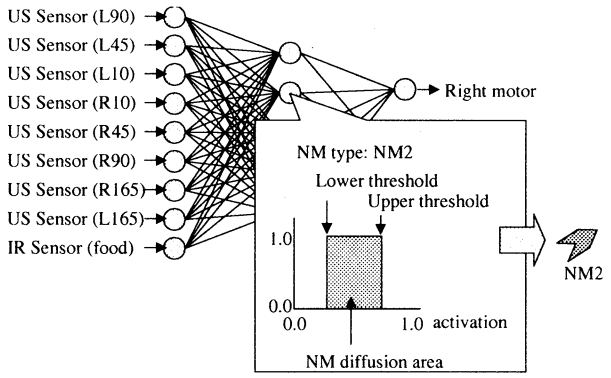


Figure 6: Diffusion of NMs.

4.2.2 Reaction to NMs

As in Figure 7, each synapse has receptors for the diffused NMs, and the received NMs cause genetically-determined synaptic modification. In the followings, we set the number of NM types to four. In this case each synapse has four kinds of NM receptors (see Figure 7).

Once the NMs are diffused, each synaptic weight is modified according to its genetically-determined interpretation as:

$$w_{ij}^{t+1} = w_{ij}^{t} + \eta \cdot R_{ij}(NMs) \cdot a_i \cdot a_j, \qquad (3)$$

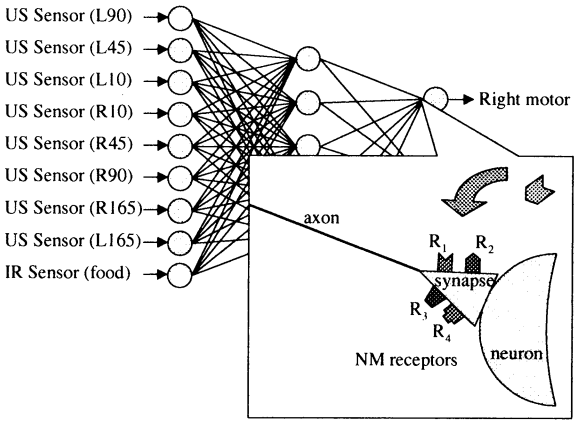


Figure 7: Receptor of NM on a synapse.

where, η is the leaning rate, $R_{ij}(NMs)$ is the parameter which determines how the synapse concerned interprets its received NMs and modify its weight. Let us explain this genetically determined interpretation on synapses in detail.

Table 2 represents an example of the receptor interpretation. Due to the number of NM types (i.e. four), all possible combination of the received NMs is sixteen, in this case. R_{ij} can take one of three values; -1, 0, 1, corresponding to *Hebbian learning* (1), *Anti-Hebbian learning* (-1), and *non-learning* (0), respectively. One of these values is assigned to each combination, and this assignment is determined through the evolutionary process.

Table 2: An example of the receptor parameters.

NM_1	NM_2	NM_3	NM_4	$R_{ij}(NMs)$	
0*	0	0	0	1	(Hebbian)
0	0	0	1	-1	(Anti-Hebbian)
...	...	...	...	...	
1**	1	1	1	0	(Non-learning)

* 0 means NM_n is not received.
** 1 means NM_n is received.

4.3 Evaluation Criterion

As mentioned in the previous subsection, the following parameters listed in Table 3 are determined by GA. These values are encoded into one-dimensional binary bit-string (chromosome).

Each individual is allowed to move during 1000

Table 3: Parameters to be searched through the evolutionary process.

θ_i	Bias input of neuron i
NM_i^{trans}	Type of NMs diffused from neuron i
θ_i^{upper}, θ_i^{lower}	Threshold value for the NM diffusion of neuron i
$R_{ij}(NM_s)$	Receptor parameter of synapse (i,j)
η	Learning rate

time-step and tested in the four kinds of the environments. The evaluation criterion used in this study is:

$$fitness = \frac{\sum_{n=1}^{4} \#\ of\ obtained\ foods\ (Env.n) \times 100}{4} \quad (4)$$

The parameters used in the following simulations are listed in Table 4.

Table 4: Simulation conditions.

α (sigmoid function)	2.0
# of NMs type	4
# of foods in each environment	100
# of population	100
# of generation	500
Reproduction ratio	30 %
Crossover ratio	30 %
Mutation ratio	30 %

4.4 Simulation Results

4.4.1 Evolvability

To verify the evolvability of our approach, we compared with the conventional ER approach, where the synaptic weights and neuron's bias are determined through the evolutionary process.

Figure 8 shows the comparison of the fitness transition of the best agent between the conventional and the proposed approach. From the figure, it is understood that the proposed method shows high evolvability.

4.4.2 Adaptability

In order to cope with the dynamically changing environment, adaptability is crucial for autonomous

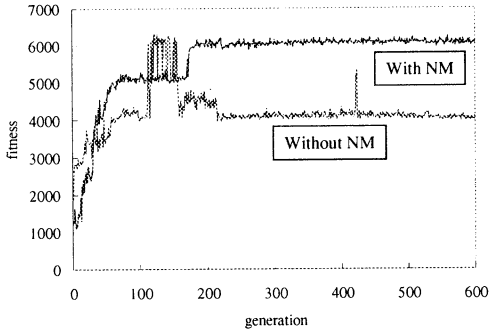


Figure 8: Comparison between the conventional and the proposed approach.

agents. Here we use the term adaptability to mean the adaptation ability to environmental changes. We applied the best individuals the obtained in Figure 8 to the environment shown in Figure 9. As in the figure, this environment is consisted of the combination of the four environments.

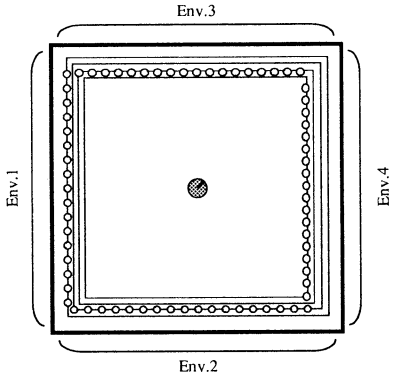
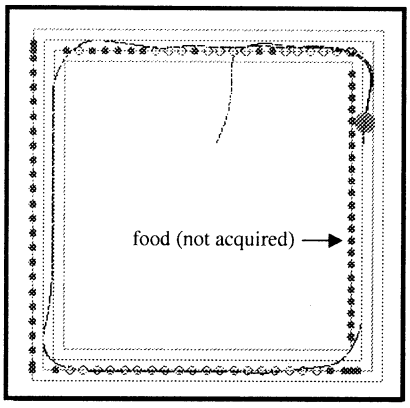


Figure 9: Environmental setup for the test of adaptability.

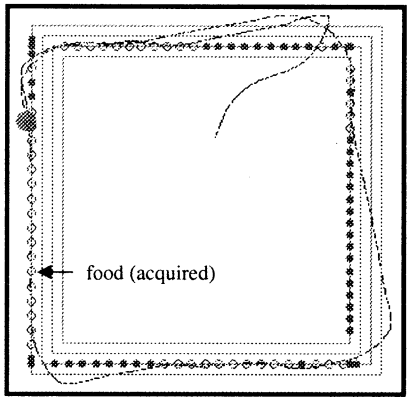
Figure 10 represents the comparison of the obtained trajectories of both methods. From the figure, it is observed that unlike the conventional approach, in our proposed approach the robot tries to adjust its movement according to the distance between the food and wall. Figure 11 shows the transition of some of the synaptic weights under the condition of Figure 10. As in the figure, some of the synaptic weights are dynamically changing. We would speculate the robot copes with this environment by dynamically yet appropriately correlating the obtained sensory readings.

5 Conclusion and Further Work

In this paper, we proposed a new adaptive controller for autonomous mobile robots by incorporat-



Without NM



With NM

Figure 10: Comparison of the trajectories.

ing the concept of dynamic rearrangement of neural networks with the use of neuromodulators. We investigated the evolvability and adaptability of the proposed method in comparison with the conventional ER approach by carrying out simulations. The obtained results are encouraging. Detail analysis of the obtained neural network (e.g. how does the NMs correlate the connections among neurons) are currently under investigation.

We are now constructing a real experimental mobile robot for experimental verification. And we foresee this approach is promising to reduce the gap between the simulations and real experiments, and to construct controllers that can elicit non-trivial tasks.

Acknowledgements

This research was supported in part by a grant from the Japanese Ministry of Education (No. A:09750487).

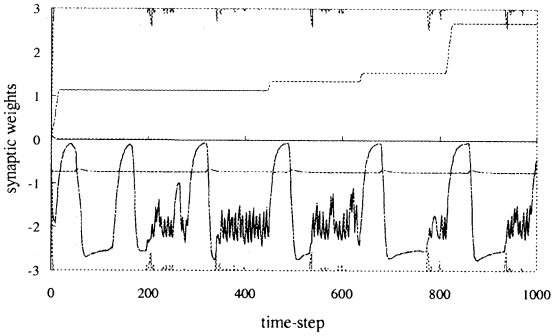


Figure 11: Transition of the synaptic weights in the obtained NM-based neural network.

References

[1] R. Beer, J. Chiel, and L. Sterling "An artificial insect", *American Scientist*, Vol. 79, pp.444-452, 1989.

[2] D. Ackley, M. Littman, "Interactions Between Learning and Evolution", C. Langton et al., *Artificial Life II*, Addison-Wesley, 1992.

[3] D. Floreano, F. Mondada, "Automatic Creation of an autonomous agent: Genetic evolution of a neural-network driven robot", *Proc. of the 3rd International Conference on Simulation of Adaptive Behavior*, 1994.

[4] S. Nolfi, D. Parisi, "Learning to adapt to changing environments in evolving neural networks", *Adaptive Behavior*, Vol. 5, No. 1, pp.75-98, 1997.

[5] P. Meyrand, J. Simmers and M. Moulins, "Construction of a pattern-generating circuit with neurons of different networks", *NATURE*, Vol. 351, 2MAY, pp.60-63, 1991.

[6] T. Kondo, A. Ishiguro, M. Takeuchi, Y. Uchikawa and P. Eggenberger, "Behavior Control of Autonomous Robot Using Dynamic Rearrangement of Neural Network", *Proc. of the 16th Annual Conf. of the Robotics Society of Japan*, Vol. 1, pp.587-588, 1998 (in Japanese).

[7] T. Kondo, A. Ishiguro, Y. Uchikawa and P. Eggenberger, "Behavior Control Using Neural Networks with Dynamic Rearrangement Function", *Proc. of FAN Symposium '98 in Fukuoka*, pp.13-16, 1998 (in Japanese).

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Maintenance of Diversity by means of Thermodynamical Selection Rules for Genetic Problem Solving

Hajime KITA[1], Naoki MORI[2] and Yoshikazu NISHIKAWA[3]
[1]Interdisciplinary Graduate School of Science and Engineering,
Tokyo Institute of Technology,
Nagatsuta 4259, Midori, Yokohama 226-8502, JAPAN
e-mail: kita@dis.titech.ac.jp
[2]School of Engineering, Osaka Prefecture University
[3]Faculty of Information Science, Osaka Institute of Technology

Abstract: For successful applications of genetic algorithms (GAs), control of convergence that achieves a good balance between exploration and exploitation is required. For this purpose, the authors have proposed a genetic algorithm with a novel selection operator called 'the thermodynamical genetic algorithm (TDGA).' In the TDGA, diversity of the population is measured by entropy, and the population in the next generation is formed by selecting individuals so as to minimize 'the free energy.' Thus, diversity is maintained systematically and explicitly. The TDGA has been applied successfully not only to optimization problems but also to problems of multi-objective optimization and those of adaptation to changing environments.
Keywords: genetic algorithms, maintenance of diversity, entropy, simulated annealing

1. Introduction

Genetic Algorithms (GA) developed by Holland et al. are techniques for adaptation and optimization inspired by the heredity and evolution of the living systems[1]. They attract attention as methods for combinatorial and nonlinear optimization that are recognized as difficult problems with conventional techniques.

For successful applications of the GA, control of convergence achieved by selection operators is important. That is, on one hand, they are required to focus the search around the good solutions by selecting them, and on the other hand, they are also required to maintain sufficient diversity to guarantee generation of novel search points by the crossover and mutation operators.

On this issue, the authors have developed a method called 'the thermodynamical genetic algorithms (TDGA)'[2]. In the TDGA, the diversity of population is evaluated explicitly as entropy and it is controlled systematically through a selection mechanism based on the principle of minimal free energy. The TDGA achieves maintenance of diversity explicitly and systematically, and it is applicable to multi-objective optimization and adaptation to changing environment as well as combinatorial optimization. The present paper sketches the concept, algorithm and applications of the TDGA.

2. Concept of TDGA

In a system at thermal equilibrium, the probability $p(x)$ that the system stays at $x \in X$ follows the Gibbs distribution:

$$p(x) = \frac{1}{Z}\exp\left(-\frac{E(x)}{T}\right) \quad (1)$$

$$Z = \sum_{x\in X}\exp\left(-\frac{E(x)}{T}\right) \quad (2)$$

where E is the energy when the system takes a state x, and T is the temperature.

Further, it is known that the Gibbs distribution is the probability distribution that minimize the

following 'free energy'[4]:

$$F = \langle E \rangle - TH \quad (3)$$

$$\langle E \rangle = \sum_{x \in X} p(x)E(x) \quad (4)$$

$$H = -\sum_{x \in X} p(x) \log p(x) \quad (5)$$

where $\langle E \rangle$ is the mean energy, and H is the entropy. We call this fact 'the principle of the minimal free energy.'

This principle can be interpreted from a viewpoint of the GA as follows: The first term of the RHS of Eq.(3) is a term to pursue the minimization of the objective function (or maximization of the fitness function in the terminology of the GA), and the second term is a term to pursue the maintenance of diversity of the population. These two goals are balanced with a parameter T.

Based on this interpretation, in the TDGA, selection of individuals that survive in the next generation is carried out so as to minimize the free energy of the population. As well as the simulated annealing[3], by gradually decreasing the temperature T, the search is focused around the good individuals found in the search process.

3. Algorithm of TDGA

A prototype algorithm of the TDGA is as follows:

Parameters

- N_{p}: Population size.
- N_{g}: Maximal generation.
- $\mathcal{T}(t)$: Annealing schedule of temperature T.

Algorithm

1. Initialize the generation counter $t = 0$, and generate initial population $\mathcal{P}(0)$ randomly.
2. Let $T = \mathcal{T}(t)$.
3. Get $\mathcal{P}_{\mathrm{O}}(t)$ by applying crossover and mutation to $\mathcal{P}(t)$, get $\mathcal{P}_{\mathrm{P}}(t)$ by applying mutation $\mathcal{P}(t)$, and let e be the elite of $\mathcal{P}(t)$.

   Let $\mathcal{P}'(t) = \mathcal{P}_{\mathrm{O}}(t) \cup \mathcal{P}_{\mathrm{P}}(t) \cup \{e\}$
4. Let $i = 1$, and $\mathcal{P}(t+1) = \phi$.
5. Let $\mathcal{P}(t+1, i, h) = \mathcal{P}(t+1) \cup \{h\}$

   where h denotes the h-th individual of $\mathcal{P}'(t)$.

   For all $h = 1 \sim 2N_{\mathrm{p}} + 1$, evaluate the free energy F of $\mathcal{P}(t+1, i, h)$, and find the individual $h_{\min}$ that minimizes the free energy:

   $$h_{\min} = \arg\min_{h} F(\mathcal{P}(t+1, i, h))$$

   Add $h_{\min}$ to $\mathcal{P}(t+1)$:

   $$\mathcal{P}(t+1) = \mathcal{P}(t+1) \cup \{h_{\min}\}$$
6. Let $i = i + 1$. If $i < N_{\mathrm{p}}$, go to Step 5.
7. Let $t = t + 1$. If $t < N_{\mathrm{g}}$, go to Step 2, otherwise terminate the algorithm.

In the above, the entropy must be evaluated considering the fact that a practical population size is much smaller than the number of the possible states of the system $|X|$. Hence we evaluate the entropy H approximately as sum of the entropies in each locus. That is,

$$H \simeq \sum_{i} H_i \quad (6)$$

$$H_i = -\sum_{j} p_{ij} \log p_{ij} \quad (7)$$

where H_i is the entropy in locus i, p_{ij} is the ratio of the allele j in the locus i[2].

4. Applications of TDGA

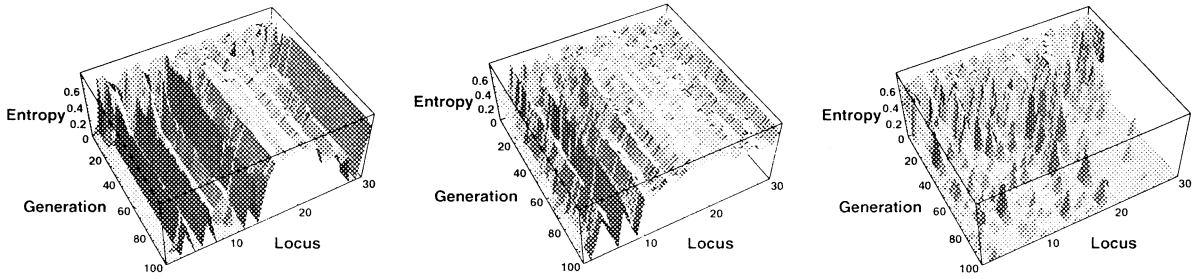
4.1 Combinatorial Optimization

To investigate the fundmental behavior of the TDGA, numerical experiments are carried out taking a knapsack problem as an example[2]. The knapsack problem is a problem of choosing items from N items each having weight a_i and value c_i so as to maximize the total value under a constraint that the total weight should not exceed a limit b. That is:

$$\max_{x_i \in \{0,1\}} \sum_{i=1}^{N} c_i x_i \quad (8)$$

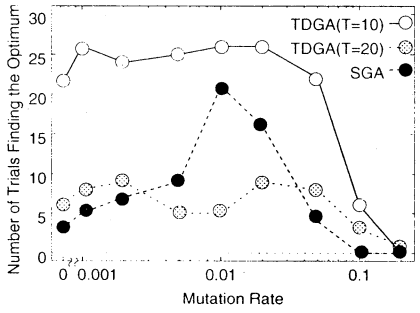
$$\text{Subject to} \quad \sum_{i=1}^{N} a_i x_i \leq b \quad (9)$$

Experiments are carried out with the following setting:

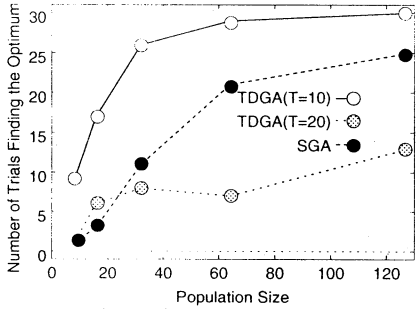


TDGA, $T = 10$ TDGA, $T = 20$ SGA

Fig. 2: Evolution of locus wise entropy. Loci are sorted by c_i/a_i.



(a) Effect of mutation rate
(Population sizes are TDGA: 32, SGA: 64)



(b) Effect of population size
(Mutation rates are TDGA: 0.02, SGA: 0.01)

Fig. 1: Comparison of TDGA with SGA using a knapsack problem.

- Size of the problem N (the number of the items): 30.
- Comparison: TDGA (with $T = 10$, 20) and Simple GA (SGA)[1].
- Sensitivity analysis: population size and mutation rate.

Number of trials that found the optimum in 30 trials with different seed for random numbers is shown in Fig. 1. and evolution of locus wise entropy in each algorithm is shown in Fig. 2. Figure 1 shows that with adequate temperature ($T = 10$), TDGA successfully find the optimum with high probability less depending on mutation rate, and with fewer population size than SGA. Figure 2 shows that such good performance of TDGA is brought about by maintaining the diversity especially in the loci of middle c_i/a_i values that need intensive search.

The authors also have proposed an adaptive annealing schedule of the temperature, and have applied TDGA to the traveling salesman problems[5].

4.2 Multi-Objective Optimization

Multi-objective optimization problems (MOP) are problems to optimize several objective functions simultaneously. Generally, it is impossible to minimize all the objective function, and existence of trade-off among objective values is intrinsic characteristics of MOP. As a rational solution for MOP, the following concepts are commonly used:

Pareto Optimal Solution: A solution that any objective value cannot be improved without degradation of some objective values. It is also called non-dominated solution.

Generally, the Pareto optimal solution is not unique. We call a set of all the Pareto optimal solutions the Pareto optimal set. Hence, requirement for methods to solve the MOP is "to find the Pareto optimal set, or if it is infinite, to sample Pareto optimal solutions that characterizes

the Pareto optimal set well by e.g. uniform sampling." Utilizing the population based search of the GA, methods of finding many Pareto optimal solutions in parallel by the GA have been studied[6].

In the TDGA, by keeping temperature T positive, the diversity of the population is maintained constantly. Hence, we can construct a method of multi-objective optimization combining the TDGA with the Pareto-based ranking technique proposed by Goldberg[1]. That is, the rank is used as the energy function of TDGA and select individuals by keeping temperature at some positive level[7]. Since the Pareto-based ranking is a well normalized fitness function and entropy is evaluated in the genotypic space, the multi-objective optimization with TDGA has an advantage that adjustment of the parameter (temperature) is easier than parameter adjustment in other techniques such as the fitness sharing[1].

4.3 Adaptation to Changing Environment

In engineering, it is highly required that the systems adapt autonomously to changing environments. To cope with environmental changes, the system must detect the change with some trial-and-error within an allowable range of performance degradation, and find a suitable solution to the novel situation.

In applications of the GA to such problems, maintenance of diversity of the population is an intrinsic requirement for continuous search. The TDGA meets this requirement well, and it has been applied to the problems of adaptation to changing environment[8]. Further, the authors have also proposed a method of controlling the temperature with a feedback technique so as to keep the entropy at a prespecified level[10], and for cases that environmental change has some recurrent nature, a method of combining the TDGA with a memory-based approach to keep the results of adaptation in past and utilize them[9].

5. Conclusion

Thermodynamical Genetic Algorithms proposed by the authors control convergence of the GA systematically. It makes applications of the GA to problems of optimization, multi-objective optimization and adaptation to changing environments more efficient.

This research was supported by the Grant-in-Aid for Scientific Research on the Priority Area 'System Theory of Function Emergence', from the Ministry of Education, Science, Sports and Culture of Japan, and by "The Research for the Future" Program, (JSPS-RFTF96I00105) of The Japan Society for the Promotion of Science.

References

[1] D. Goldberg: Genetic Algorithms in Search, Optimization, and Machine Learning, *Addison-Wesley* (1989).

[2] N. Mori, J. Yoshida, H. Tamaki, H. Kita and Y. Nishikawa: 'A Thermodynamical Selection Rule for the Genetic Algorithm,' *Proc. IEEE ICEC'95*, pp. 188-192 (1995).

[3] S. Kirkpatrick et al.: Optimization by simulated annealing, *Science*, Vol. 220, pp. 671-680 (1983).

[4] Fukao: *Thermodynamical Theory of Distributed System*, Shokodo (1987, in Japanese).

[5] K. Maekawa, N. Mori, H. Tamaki, H. Kita and Y. Nishikawa: 'A Genetic Solution for the Traveling Salesman Problem by Means of a Thermodynamical Selection Rule,' *Proc. IEEE ICEC'96*, pp. 529-534 (1996).

[6] H. Tamaki, H. Kita and S. Kobayashi: 'Multi-Objective Optimization by Genetic Algorithms: A Review,' *Proc. IEEE ICEC'96*, pp. 517-522 (1996).

[7] H. Kita, Y. Yabumoto, N. Mori and Y. Nishikawa: 'Multi-Objective Optimization by Means of the Thermodynamical Genetic Algorithm,' *Proc. PPSN IV*, pp. 504-512 (1996).

[8] N. Mori, H. Kita and Y. Nishikawa: 'Adaptation to a Changing Environment by Means of the Thermodynamical Genetic Algorithm,' *Proc. PPSN IV*, pp. 513-522 (1996).

[9] N. Mori, S. Imanishi, H. Kita and Y. Nishikawa: Adaptation to Changing Environments by Means of the Memory Based Thermodynamical Genetic Algorithm, *Proc. ICGA97*, pp. 299-306 (1997).

[10] Naoki Mori, Hajime Kita and Yoshikazu Nishikawa: Adaptation to a Changing Environment by Means of the Feedback Thermodynamical Genetic Algorithm, *Proc. PPSN V*, pp. 149-158 (1998).

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Protein Folding by A Hierarchical Genetic Algorithm

Osamu Takahashi, Hajime Kita, and Shigenobu Kobayashi

Dept. of Computational Intelligence & Systems Science, Interdisciplinary Graduate School of Sci. & Eng.,
Tokyo Institute of Technology, 4259, Nagatsuta, Midori-ku, Yokohama, 226-8502, Japan.
E-mail: osamu@fe.dis.titech.ac.jp, kita@dis.titech.ac.jp, kobayasi@dis.titech.ac.jp

Abstract

The protein folding problem is a important problem in molecular biology. It can be formulated as a minimization problem on an energy function. However, it is very hard because of numerous local minima. To solve it, in this paper, we present a hierarchical Genetic Algorithm (GA) combining the trinary digit (*trit*) string GA and the real-coded GA. The proposed method has two layers. At the upper layer, an optimal sub-space, which is defined by designated borders of dihedral angles, is searched by the trit-string GA. At the lower layer, a minimum energy value in a given sub-space is searched by the real coded GA. To evaluate its effectiveness, it was applied to a folding problem of [Met]-enkephalin. As a result, the method succeeded in finding out two types of global structures with energy values that are lower than the known one.

Keywords: protein folding, hierarchical GA, real-coded GA, trinary digit (trit) string GA, minimum-energy structure, enkephalin

1. Introduction

The prediction of protein tertiary structure from only its amino-acid sequence information as a primary structure is known as the *protein folding problem*. It is a important problem in molecular biology. It can be formulated as a minimization problem on an energy function. However, it is very hard because of numerous local minima.

Simulated annealing method[4], Monte Carlo with Minimization (MCM)[2,3] and Multicanonical Algorithm[5,6] as an improved Metropolis method have been applied to this problem. On the other hand, several approaches based on genetic algorithms[1] also have been applied to this domain. But most of them researched effectiveness of GAs not in real protein folding problems, but in artificial grid world models[7,8]. Recently, hybrid genetic algorithms[9] have been proposed which incorporate gradient-based minimization. That approach improves the search performance on real problems with a local search method.

For the protein folding problem, we use the dihedral angles representation. Each dihedral angle can be divided into three regions, having local minima considering the local structure of protein, as shown in Fig.1. Taking this into consideration, we propose a hierarchical GA. That is, for global search, we divide the whole search space into partial sub-spaces, and replace the optimization problem in the continuous space with a combinatorial one. The upper layer of the hierarchical GA searches a sub-space which includes the global minimum-energy structure. The fitness of each sub-space is defined as the minimum energy value in the sub-space is searched. Hence we adopt the GA using floating point representation, i.e. real-coded GA[10,11] as the lower layer GA. Though the searching space is divided, it may have multiple minima because of global interaction of the folding protein. Therefore, searching methods applied to the sub-space must be robust.

Numerical experiments show that this method can solve penta-peptide energy minimization problem from the whole search space.

2. Protein Folding Problem

The prediction of protein tertiary structure (a 3D shape) from its primary structure (a sequence of amino-acids) is a fundamental task in molecular biology. This problem is commonly referred to as the *protein folding problem*. Efforts to solve it nearly always assume that the native conformation corresponds to the global minimum of the free energy of the molecule. Given this assumption, it is necessary in solving the problem to develop efficient global minimization techniques. However, it is very hard to solve because of non-linear and multi-modal nature of the energy surface.

Assuming fixed bond lengths and bond angles, a protein tertiary structure could be represented only by the dihedral angles, that are rotational angles between atoms which make covalent bond. Each rotatable bond usually has three minimum-energy angles ($\pi, \pm\pi/3$) and three maximum energy angles ($0, \pm 2\pi/3$), because of sterical angle of bonds (Fig. 1).

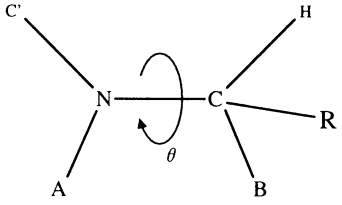


Fig. 1 Illustrative sample sterical angle. Atoms N, H, B and R form a regular tetrahedron. The energy becomes high when the atoms A and B are at interference position.

We use an energy function ECEPP/2[13] as an objective function to be minimized. This function is popularly used in protein conformation prediction study, and therefore it is suitable for evaluating the effectiveness of optimization methods.

ECEPP/2, as empirical potential energy functions, calculates electrostatic energy E_c, nonbonded energy E_{LJ} using a modified Lennerd-Jones 6-12 potential, any interaction between designated donor and acceptor atoms as a hydrogen bond energy E_{HB}, torsional energy E_{tor} that are computed for all peptide bond (ω) dihedral angles, and for designated side-chain (χ) dihedral angles and end group dihedral angles, and not for (ϕ, φ). Then the total energy E_{tot} are given as summation of these energy terms.

$$
\begin{aligned}
E_c &= \sum_{(i,j)} \frac{332.0 q_i q_j}{D r_{ij}} \\
E_{LJ} &= \sum_{(i,j)} \left(\frac{A_{ij}}{r_{ij}^{12}} - \frac{B_{ij}}{r_{ij}^{6}} \right) \\
E_{HB} &= \sum_{(i,j)} \left(\frac{C_{ij}}{r_{ij}^{12}} - \frac{D_{ij}}{r_{ij}^{10}} \right) \\
E_{tor} &= \sum_i U_i \left(1 \pm \cos\left(n_i \Theta_i\right)\right) \\
E_{tot} &= E_c + E_{LB} + E_{HB} + E_{tor}
\end{aligned}
\qquad (1)
$$

where r_{ij} is the distance between atom i and atom j in Å, q_i is the partial electric charge of atom i in electric charge units, Θ_i is the value of the dihedral angle, and n_i gives the symmetry of the barrier. Please refer ECEPP/2[13] for the factors 332.0, $A_{ij}, B_{ij}, C_{ij}, D_{ij}$ and U_i. Here we use empirical value 2.0 for the dielectric constant D.

The polypeptide we used in our calculations is [Met]-enkephalin (H-Tyr-Gly-Gly-Phe-Met-OH), as shown in Fig.2, which is popular in *ab initio* protein structure prediction studies[2,3,5,9].

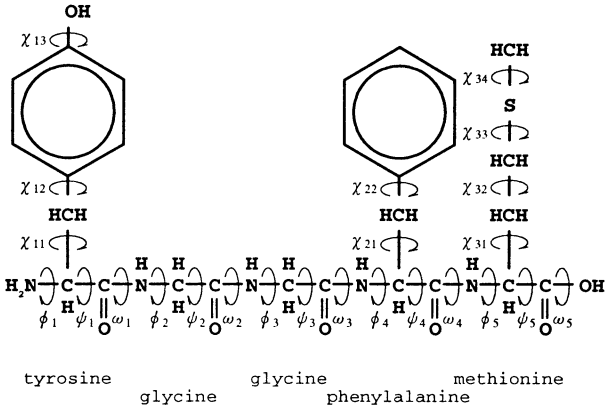


Fig. 2 Illustrative drawing of the [Met]-enkephalin molecule.

3. Hierarchical Genetic Algorithm

In this paper, we propose a solving procedure that is composed of two layers. Considering the characteristics of the energy function stated in the previous section, we divide the dihedral angles at $0, \pm 2\pi/3$. Then the whole search spaces are divided into 3^{24} sub-spaces.

The upper layer searches a sub-space which includes the global minimum-energy structure. The fitness of each sub-space is defined as the minimum energy value in the sub-space. It is searched by the lower layer GA. That is, it behaves as a local search engine to find the minimum-energy value in the corresponding sub-space. Thus the proposed hierarchical search method treats optimization in the continuous search space as a kind of combinatorial optimization.

3.1 The Upper Layer GA

Dihedral angles in a polypeptide have three local energy barriers at corresponding atoms interfering position, as shown in Fig.1. Then, we use *trinary digit* ($x_i \in \{1,2,3\}$) gene to a represent dihedral angle (θ_i).

$$
x_i = \left\lfloor \theta_i / \left(\frac{2\pi}{3}\right) \right\rfloor \quad \left\{ \theta_i \,\middle|\, \left(\frac{2\pi}{3}\right) \le \theta_i < \left(\frac{8\pi}{3}\right) \right\} \qquad (2)
$$

We considered an appropriate crossover operation based on the polypeptide structure to search combinatorial protein structures. In a polypeptide

structure, dihedral angles in main-chain are more important than those in side-chain, and those in terminals are less important than others. Therefore, we design a chromosome structure that has the same adjacency relationship with the objective polypeptide. Using this structure, the crossover operation may keep a block of genes which should be inherited. We extend multi-point crossover to match the branching structured chromosome. The crossover operator sets n cross over points on the chromosome, then swap regions, that enclosed by cross-points, of two parents, as shown in Fig.3.

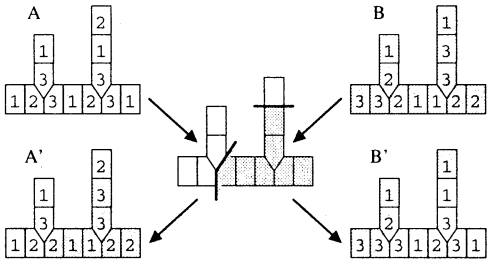


Fig. 3 A part of sample representation chromosome of a polypeptide and multi-point crossover.

As a generation alternation model, we employ the Minimal Generation Gap (MGG)[12] model, shown in Fig.4. In this model, a generation alternation is done by applying the crossover n times to a pair of parents randomly chosen from the population. From the parents and their children, we select the best individual and a random one using the roulette wheel technique. With them the original parents are replaced.

The initial population is randomly generated, and no mutation operation is used.

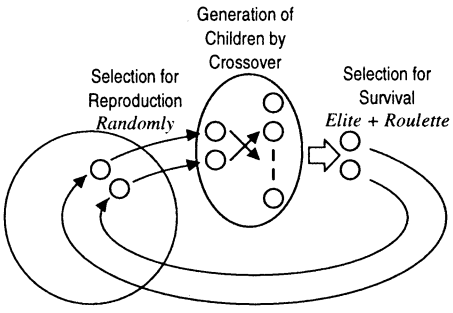


Fig. 4 Minimal Generation Gap (MGG)[12] model as a generation alternation model for the Upper Layer GA

3.2 The Lower Layer GA

To find the minimum value in a given sub-space, we use the real-coded Genetic Algorithm, where chromosome is a real number vector consisting of the dihedral angles (ϕ_1, φ_1, ω_1, ..., χ_{54}). Ono et al. have proposed the Unimodal Normal Distribution Crossover (UNDX)[10] for real-code GAs. The UNDX generates children obeying a normal distribution around the parents as shown in Fig.5. The normal distribution is centered at the midpoint of parents 1 and 2, and having a standard deviation proportional to the distance a parent from the midpoint in the direction of the line connecting the parents. Further, the standard deviations in the perpendicular directions are proportional to the distance of the third parent and the lines connecting the parent 1 and 2.

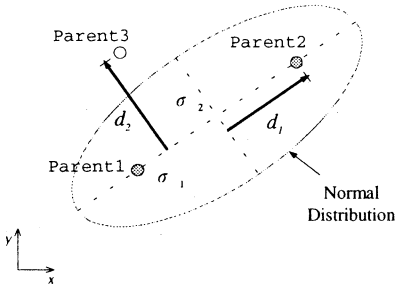


Fig. 5 Unimodal Normal Distribution Crossover (UNDX)[10]

It has been shown that the UNDX can efficiently optimize multimodal and highly epistatic functions. We believe that the reasons of effectiveness of the UNDX are that the UNDX searches globally in the early phase of search where parents are scattered all over the search space and locally in later phase where parents gather in some promising areas, less depending on the coordinate system. For the lower layer GA, we use the multi-parental extended version of the UNDX called UNDX-m[11] with m=2. This extension enhances the search ability of the UNDX on complex surfaces of fitness functions.

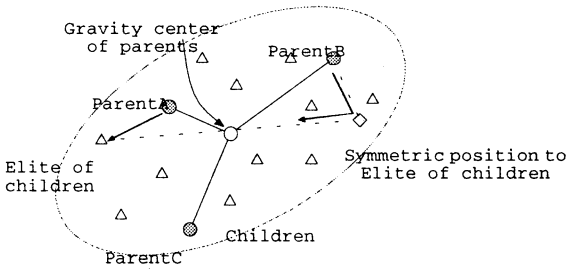


Fig.6 Distance Dependent Alternation (DDA) as a generation alternation for the Lower Layer GA

A generation alternation model used with the UNDX-m is required to keep diversity of parents, because the UNDX-m tends to search region near the parents' gravity center that causes shrinking of

population. Here, we propose the Distance Dependent Alternation (DDA) as a generation alternation scheme for keeping diversity of population. First, like the MGG, m+2 parents are randomly selected from the population. Second, with the UNDX-m, the selected parents generate several children.

Then, as shown in Fig.6, the DDA selects the elite from the children and finds the parent nearest to it. If the elite child is superior to the parent, the parent is replaced with the child. Otherwise, the procedure finds the parent nearest to the *symmetric* point of the elite child with the gravity center as the symmetric center. Then, the parent is compared with the elite child. If it is inferior, it will be replaced with the elite child.

4. Experiments and Results

We applied the proposed method to an energy minimization problem of [Met]-enkephalin where energy is calculated through ECEPP/2. We executed the proposed method three times with the same parameters except for seeds for random numbers. We set a population size to be 100, the number of crossovers to be 30 and the number of crossover-points to be 5 for the upper layer GA, a population size to be 50 and the number of crossovers to be 20 for the lower layer GA. At the upper layer, we fixed dihedral angles ωs at sub-space 1. At the lower layer, we set the initial population scattering around the center of sub-space by standard deviation of 10 degrees for dihedral angles ωs and 40 degrees for the other dihedral angles ϕs, φs and χs.

After several hundreds generations, we obtained the same partial sub-space including the structure almost same with the known[3] best one, as shown in Fig.7, in all the trials.

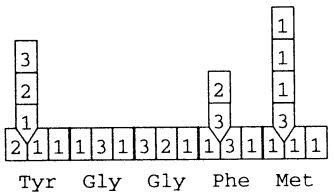


Fig. 7 The partial sub-space including the structure almost same with the known[3] best one.

We call this structure the Type I structure, shown in Fig.9. Further, the proposed algorithm found out another local optimum structure called the Type II structure, shown in Fig.10, at the same time. These two structures are summarized in Table 1. The energy value of the Type II structure is quite near to that of Type I, but its tertiary structure is different.

The success probability of UNDX-2 with DDA of finding out the global minimum-energy structure in the corresponding partial sub-space is reached to about 0.8. However, it requires several thousand generation to converge. The total number of evaluation times amounts to $10^{8\sim9}$, and one execution of the proposed method spends a few days using ten PCs whose CPUs are Pentium-Pro 200MHz.

Dihedral angle (degrees)								
		main chain			side chain		
		Type I	Type II	known[3]	Type I	Type II	known
Tyr	ϕ_1	-86.12	-174.41	-86			
	ψ_1	155.77	-24.75	156			
	ω_1	-176.84	-177.61	-177			
	χ_{11}				-173.03	70.19	-173
	χ_{12}				-101.26	-82.04	79
	χ_{13}				23.22	150.58	-166
Gly	ϕ_2	-154.19	130.93	154			
	ψ_2	82.42	-157.33	83			
	ω_2	169.73	179.84	169			
Gly	ϕ_3	82.47	-82.50	84			
	ψ_3	-75.04	-63.54	-74			
	ω_3	-169.47	-177.62	-170			
Phe	ϕ_4	-134.49	-136.01	-137			
	ψ_4	18.58	19.12	19			
	ω_4	-174.28	-174.41	174			
	χ_{41}				58.85	59.11	59
	χ_{42}				-85.92	94.59	-85
Met	ϕ_5	-162.74	-164.72	164			
	ψ_5	161.21	-19.63	160			
	ω_5	-179.72	177.87	-180			
	χ_{51}				52.72	55.15	53
	χ_{52}				175.23	178.08	175
	χ_{53}				-180.00	179.81	-180
	χ_{53}				-178.63	-178.81	-59
Energy (kcal/mol)		-16.087	-15.468	-15.205** -12.904*			

** is re-calculated by our energy function which is converted to C program using double precision with latest parameters.
*is original[3] energy value.

Table 1. Two types of minimal-energy structure obtained by the proposed method, and reference structure[3]. The angles correspond to Fig.2.

5. Discussions

For visualization of the behavior of the upper level GA, we define a distances between sub-spaces A and B as follows:

$$d_{AB} = \sum_i k*|a_i - b_i| \quad \begin{cases} k = 1 & if\ i \in main\,chain \\ k = 0.1 & if\ i \in side\,chain \end{cases} \quad (3)$$

where a_i and b_i are genes at locus i of sub-space A and B respectively.

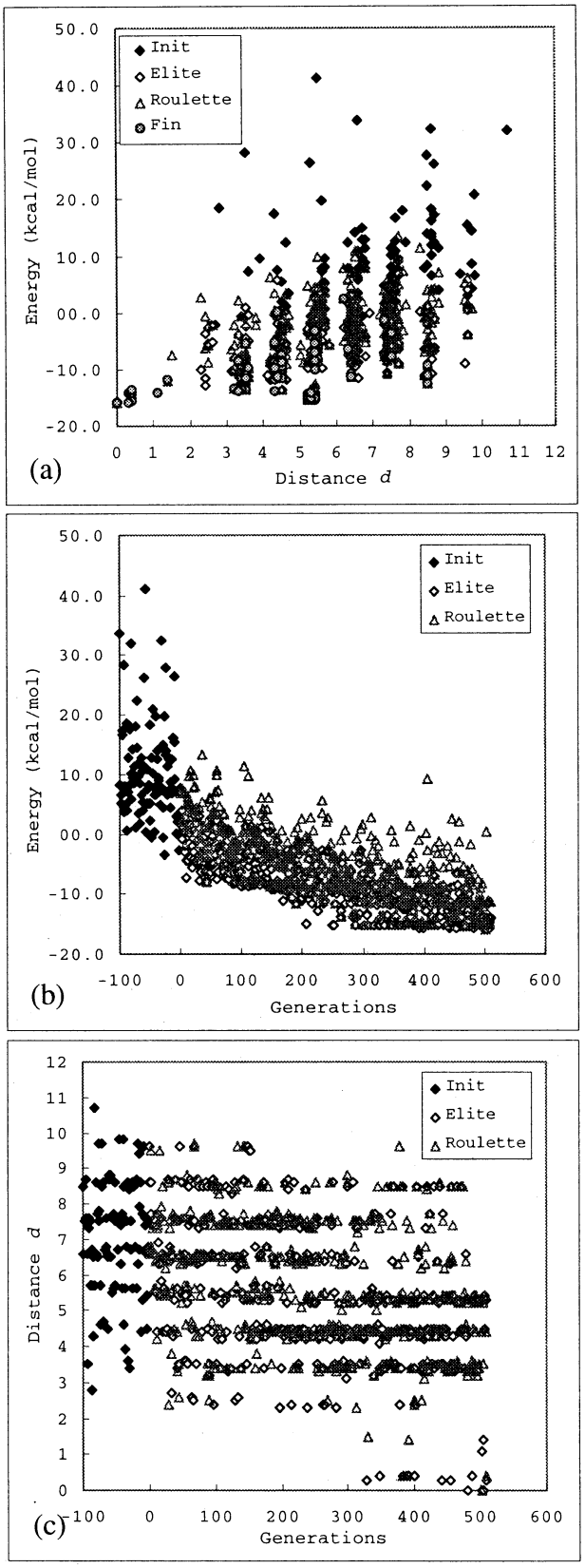
Fig.8(a) shows the relation between the energy and the distance d for all individuals generated through one execution, where d means the distance between each individual and Type I sub-space. The distance between Type I and Type II is 5.2. The group of individuals with lower energy around d=0 and 5.2 correspond to Type I and Type II respectively. Fig.8(b) shows the relation between the energy and the generation for all individuals. Fig.8(c) shows the relation between the distance d and the generation for all individuals. A remarkable point in Fig.8(c) is the distance gap between 0 and over 2 after 300 generations. It suggests that the minimum-energy structure sub-space Type I may be isolated by energy barrier in the sub-space level.

6. Conclusion & Future work

In this paper, we proposed a hierarchical GA, which is stacked the trinary digit string GA with the real-coded GA, for energy minimization problem of protein structure. We succeeded in obtaining two excellent minimal energy structures from complex energy surface at the same time in a penta-peptide example. The proposed method robustly finds out the minimal energy structure of [Met]-enkephalin, but it requires a large amount of evaluating calculations from random initial structures.

In a natural situation, the native structure of the protein may be obtained by folding along the energy valley. However, a solvent effect caused by water seems to play an important role in folding. If the water effect is omitted because of its difficulty in computation, the energy valley reaching to the minimum-energy structure may be interrupted by some barrier. Then, we conceive that searching minimal energy structures without water effect requires global combinatorial search over many energy hill.

As a future work, we would like to challenge longer protein by reducing the amount of calculations.



Init: Initial population
Elite: Elite child of each generation
Roulette: A child selected through roulette wheel selection
Fin: Final population

Fig. 8 The behavior of the Upper Layer GA

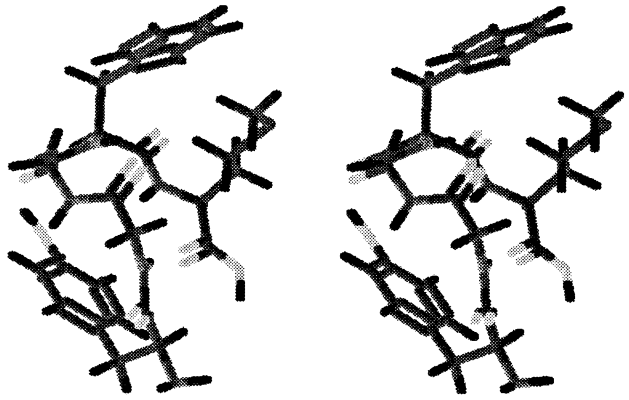


Fig. 9 Stereo view of Type I structure.
This figure was created with RasMol[14].

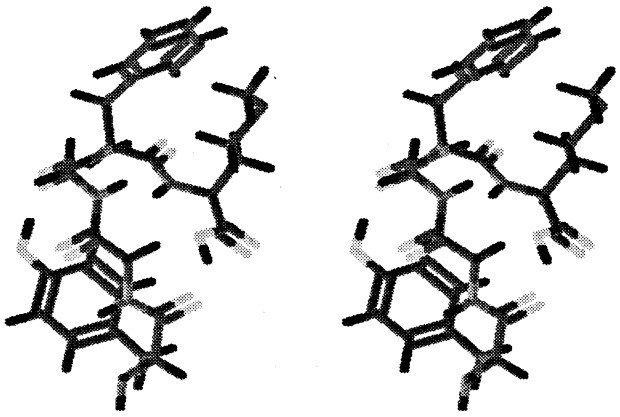


Fig. 10 Stereo view of Type II structure.
This figure was created with RasMol[14].

References

1. Goldberg DE (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley Publishing Company Inc.

2. Li Z, Scheraga HA (1987), Monte Carlo-minimization approach to the multiple-minima problem in protein folding, Proc. Natl. Acad. Sci. USA, Vol.84, pp.6611-6615, Oct. 1987 Chemistry

3. Nayeem A, Vila J, and Scheraga HA (1991), A Comparative Study of the Simulated-Annealing and Monte Carlo-with-Minimization Approaches to the Minimum-Energy Structure of Polypeptide:[Met]-Enkephalin, Journal of Computational Chemistry, Vol.12, No.5, pp.594-605, 1991

4. Kawai H, Kikuchi T, and Okamoto Y (1989), A prediction of tertiary structures of peptide by Monte Carlo simulated annealing method, Protein Engineering Vol.3, No.2, pp.85-94, 1989

5. Hansmann UHE, Okamoto Y (1997), Numerical Comparisons of Three Recently Proposed Algorithms in the Protein Folding Problem, J. of Comp. Chem., 1997, Vol.18, No.7. pp.920-933

6. Hansmann UHE, Okamoto Y (1998), Tertiary Structure Prediction of C-Peptide of Ribonuclease A by Multicanonical Algorithm, J. Phys. Chem. B 1998, 102, pp.653-656

7. Unger R, Moult J (1993), Genetic Algorithms for Protein Folding Simulations, J. Mol. Biol. (1993) 231, pp.75-81

8. Patton AL, Punch WF, and Goodman ED (1995), A Standard GA Approach to Native Protein Conformation Prediction, Proceedings of 6th Inernational Conference on Genetic Algorithms 1995, pp.574-581

9. Merkle LD, Lamont GB, Gates Jr. GH, and Pachter R (1996), Hybrid Genetic Algorithms for Minimization of a Polypeptide Specific Energy Model, Proc. of 1996 IEEE Intr. Conf. on Evolutionary Computation, pp.396-400

10. Ono I, S Kobayashi (1997), A Real-coded Genetic Algorithm for Function Optimization Using Unimodal Normal Distribution Crossover, Proceedings of 7th Inernational Conference on Genetic Algorithms 1997, pp.246-253

11. Kita H, Ono I, and Kobayashi S (1998), Multi-parental Extension of the Unimodal Normal Distribution Crossover for Real-coded Genetic Algorithms, SICE System, Information Joiunt Symposium (in Japanese).

12. Satoh H, Yamamura M, and Kobayashi S (1996), Minimal Generation Gap Model for GAs considering Both Exploration and Exploitation, Proceedings of IIZUKA'96, pp.494-497

13. Browman MJ, Carruthers LM, Kashuba KL, Momany FA, Pottle MS, Rosen SP, and Rumsey SM (1983), ECEPP/2: Empirical Conformational Energy Program for Peptides, Quantum Chemistry Program Exchange QCPE Program No.454, Indiana University, Department of Chemistry

14. Sayle RA, Milner-White EJ (1995), TIBS 1995, Vol.20, pp.374-376

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Artificial Behavior of Cell-Like Structure with Polarized Elements

†Takefumi Kohashi ‡Toshiaki Takayanagi †Keiji Suzuki †Azuma Ohuchi

†Laboratory of Harmonious Systems Engineering
Research Group of Complex Systems Engineering
Graduate School of Engineering, Hokkaido University
Kita 13, Nishi 8, Kita-ku
Sapporo, 060-8628 JAPAN

‡Section of Pathology
Institute of Immunological Science
Hokkaido University
Kita 15, Nishi 7, Kita-ku
Sapporo, 060 JAPAN

Abstract

In this paper, we try to simulate cell-like structures that can show self-assemble process or self-maintenance process and give quantitative descriptions of system. A great deal of effort has been made for simulation of the model imitating real living cells. What seems to be lacking, however, is biological preciseness. This is due to computational complexity nature of the problem. Real membranes of living cells are very complex structure that consist of proteins, lipids, and saccharides. The main ingredient of constructing membranes of living cells is lipid molecules. Owing to the properties of lipids molecules, in water, these molecules can form peculiar aggregates(e.g. micelles, bilayers). We propose the model that have properties of real membranes of living cells, and analyze a behavior of system using two types of quantitative descriptions.

1 Introduction

Regarding living creature as a system, its robustness or flexibility is superior to artificial systems by far. Therefore, numerous attempts have been made to let artificial systems taking advantage of such properties.

To begin with, we have been interested in superior properties of living creature from the view point of engineering. Particularly we are interested in self-maintenance ability of living cells. Living cells have superior ability of self-maintenance . One of the important elements that makes it possible for living cells to self-maintainance is fluidity or flexibility of membranes of living cells. Although menbranes of living cells are very complex structures, it is flexible and fluid-like. There are numerous researchs about menbranes of living cells in various fields[2][3][4]. A large number of studies also have been made on modeling of membranes of living cells. What seems to be lacking, however, is biological preciseness.

In this research, we propose a model that imitate lipid molecules(it is main ingredient of membranes) to reserch flexibility or fluidity of membranes of living cells, and observe behavior of systems. In addition, we analyze the systems using two types of quantitative descriptions.

2 Membranes of living cells

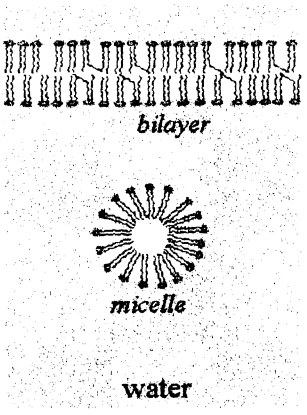


Figure 1: Example of lipid aggergates

Membrane of living cells are very complex structures that consist of proteins, lipids, and saccharides. Although proteins control many high level activity of living cells, main ingredient of membranes of living cells is lipid molecules.

A lipid molecule has a hydrophobic part and a hydrophilic part. Namely lipid molecules are amphipathic molecules. An amphipathic lipid molecules

have a polar head and a non polar tail. In water environment, these molecules can self-associates or self-assemble into small molecular aggregates such as micelles and bilayers(Fig 1).

Unlike solid or rigid particles and macromolecules such as DNA, these aggregates structures are flexible and fluid-like. This is due to the fact that the amphiphilic molecules in micelles or bilayers are held together not by strong covalent and ionic bonds, but by weaker van der waals, hydrophobic, hydrogen-bonding and screened electrostatic forces. The main forces governing the self-assembly of amphiphilic particles are believed to arise from the hydrophobic property of lipid tails at the hydrocarbon-water interface. Two opposing forces control the effective headgroup area exposed to the aqueous phase: The hydrophobic property of the lipid tails causes molecules to associate, while the hydrophilic property of the lipid headgroups tends to force the molecules to remain in contact with water.

3 Model

To realize more realistic simulation, we propose computational model that have some properties mentioned above. Our model is centered on what is believed to be the most important property of lipid molecules responsible for the formation of lipid aggregates, namely, the different hydrophobicity of the lipid's head and tail.

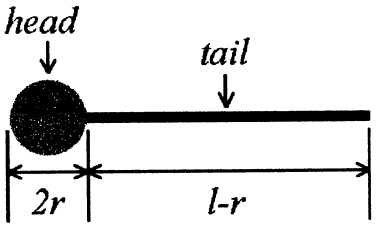


Figure 2: Basic elemt

3.1 Basic Elemens

Basic elements in our model are shown Figure 2. To better reflect this amphiphilic property, basic elements are modeled as structured particles of large haeds and long, thin tails. The head of a basic element is defined as a sphere of radius r. The tail is represented as a thin, inflexible rod of length L.

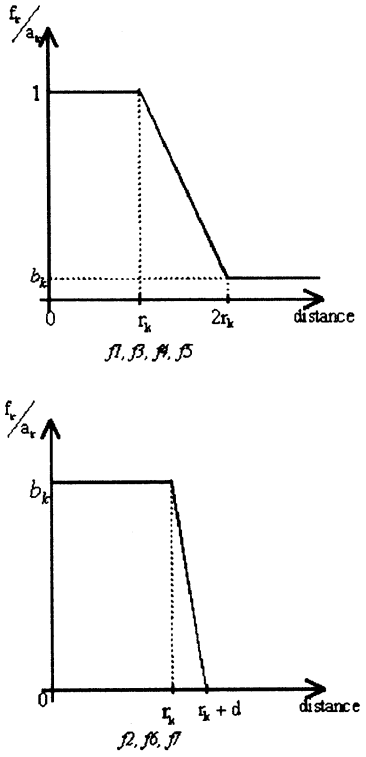


Figure 3: Two ramp functions

3.2 Forces

Based on biological properties of lipids, we define seven inter-element interacting forces. These forces were proposed by Linglan and Yun[2], and we introduce a new force that is a repulsion force between tails.

These seven forces, outlined below, are defined individual parts of any pair of elements in the system. Note that heads and tails play different roles in these definition.

- f_1: head-head attraction.
- f_2: head-head repulsion.
- f_5: tail-tail attraction.
- f_7: tail-tail repulsion.
- f_3 and f_4: tail-head and head-tail repulsion.
- f_6: repulsion force pair.

f_1, f_3, f_4, and f_5 is the force to express effect of hydrophobicity in the the lipid tails and hydrophilicity in the lipid heads. These forces have relatively long range. f_2 is the force to express effect of electrostatic

charge and hydrophilicity. This force has relatively short range. f_6,f_7 is the force to express effect of incompressibility of molecules. This force has very short range.

3.3 Calculate the forces

Magnitude of interaction is large in the close distance between the points of action, and small in the distant. To simplify calculation, we use two types of ramp functions that have property mentioned above. It was proposed by Linglan and Yun to realize effects of hydrophobicity, hydrophilicity, and electrostatic charge by combining various ramp functions[2].

One of the two ramp functions has a relatively smooth reduction of the force magnitude as the distance exceeds the given range. This function used for the long range forcesf_1, f_3, f_4,and f_5. The other ramp function, with more abrupt reduction, is used for the short range forcesf_2, f_6, and f_3. The magnitude of each force f_k is determined by the distance d and the three function parameter r_k, a_k, and b_k, where r_k is the radius of the range of force f_k with maximum magnitude, a_k is the maximum magnitude of force f_k, and b_k is the residual or minimum magnitude of force f_k.

Each forces produces a torque relative to the center of mass of the particle:

$$\tau = f_k * d$$

where d is the distance from the point of action of force f_k to the geometric center of the particle.

3.4 Function of movement

There are two types of movement for each element. Namely liner and rotational movement. The liner movement and the rotational movement of element i are determined respectively by the following:

$$F_i(t) = \left(\sum_{j \neq i} \sum_{k} f_k(t) \right) + rand_1(t) - c_1 * v_i(t)$$

where k ranges over all forces from j to i, $f_k(t)$ is the force placed on element i at time t, $rand_1$ is a random force placed on element$i$, $v_i(t)$ is the current velocity of element$i$, and $c_1 > 0$ is the friction coefficient.

$$T_i(t) = \left(\sum_{j \neq i} \sum_{k} \tau_k(t) \right) + rand_2(t) - c_2 * \omega_i(t)$$

where k ranges over all forces from j to i. $\tau_k(t)$ is the torque generated by force f_k at time t, $rand_2$ is a

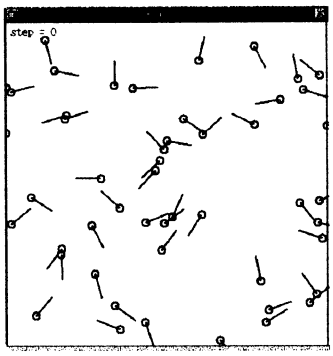


Figure 4: Initial state

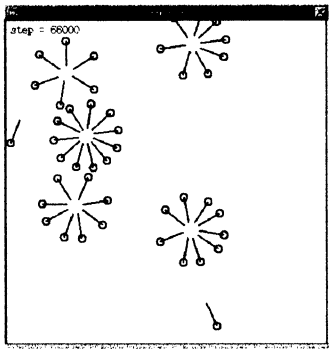


Figure 5: after 66000 steps

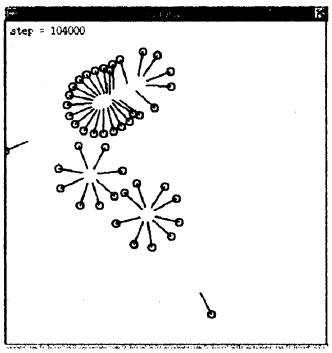


Figure 6: after 104000 steps

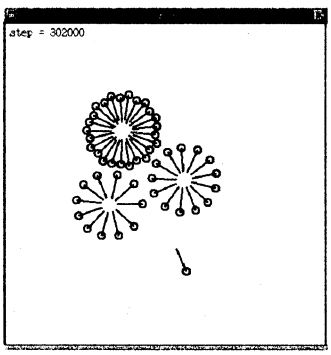


Figure 7: after 302000 steps

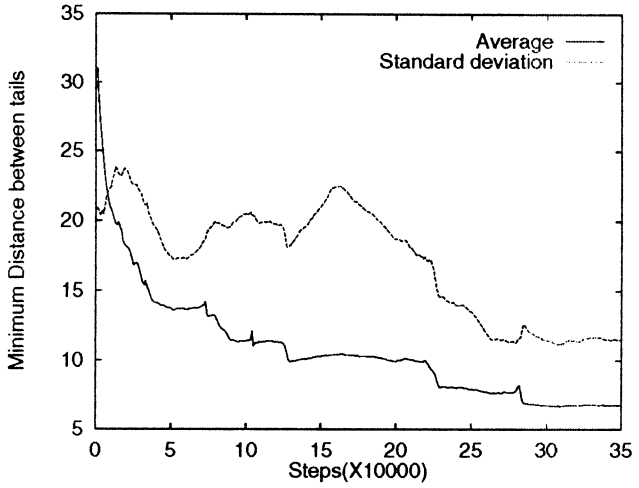


Figure 8: minimum distance between tails and its standard deviation

random torque, $\omega_i(t)$ is the current rotational velocity, and $c_2 > 0$ is rotational coefficient.

The rotational inertia I can be calculated by assuming each element to be a uniform thin rod of length $r + l$: $I = \frac{1}{12}(r + l)^2$

4 Experiments and Results

The purpose of experiment is to observe a behavior of elements, and analyze a transition of the system using two types of quantitative descriptions.

The size of the environment is 450 X 450 (normalized) units. There are 50 basic elements randomly distributed in the initial state of the system. Figure 4, 5, 6, and 7 show a process of aggregating basic elements. Finally we can observe that some cell-like structures (like micelles) are formed. System reached a relatively stable state after 310000 steps. The major effort in implementation has been the adjustment and final selection of the parameters.

Next, we give two kind of quantitative descriptions. Namely average of minimum distance between tails of each pair of elements and its standard deviation.

There are same features for all lipid aggregates. Lipid aggregates are structures that each tail part of lipid molecules avoid contacting water. The tails of lipid molecules are hydrophobic, meaning that in an aqueous environment, the tails will move toward each other, forming a structure to create an environment that does not contain water. Such structure is stable from energy.

Although we want to know energy transition to investigate system, it is impossible to define energy exactly in this system. So we noticed distance between tails of each pair of basic elements. We use the distance between tails of each pair of basic elements as an indicator of stability of system. Figure 8 shows that transition of minimum distance between tails and its standard deviation. Both average value and standard deviation drop gradually. It seems reasonable to suppose that system get into stable state. Average graph shows interesting transition. About 70000 steps, 10000 steps, and 27000 steps, there is a steep rise and a rapid drop immediately. This means as follows. When two cell-like structures unite as a new structure, stability of system is broken once, and after the union, system gets into more stable state.

5 Conclusion

We proposed a model that have properties of real membranes of living cells, analyzed system using some quantitative descriptions.

References

[1] Robert K. Murray, Daryl K. Granner, Peter A. Mayers and Victor W. Rodwell *HARPER'S BIOCHEMISTRY 23rd Edition*, Applton & Lange, 1993 (Japanese Edition: Maruzen Publishers, pp510-527, 1993)

[2] Linglan Edwards and Yun Peng, "Computational Models for the Formation of Protocell Structure", *Sixth International Conference on Artificial Life* , pp35-42, 1998

[3] Francisco J. Varela, Humberto R. Maturana, and R. Uribe, "Autopoiesis: The organization of living systems, its characterization and a model", *BioSystems*, 4:13-28, 1974

[4] Barry McMullin, and Francisco J. Varela, "Rediscovering Computational Autopoiesis" *Forth European Conference on Artificial Life* , pp38-47, 1997

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Origin and evolution of early peptide-synthesizing biomachines by means of hierarchical sociogenesis of intracellular primitive tRNA-riboorganisms

Koji Ohnishi, Shoken Hokari, and Hiroshi Yanagawa*
Dept. of Biology, Faculty of Science, Niigata University, Ikarashi-2, Niigata 950-2181, Japan,
ohnishi@sc.niigata-u.ac.jp; *Mitsubishi Kasei Institute of Life Sciences, Machida, Tokyo,194,Japan

ABSTRACT: Origin of mRNAs and genetic codes were analyzed based on poly-tRNA theory (Ohnishi (et al.),1993-98[1-8]). Tandem (kin) association of tRNAs (poly-tRNA) interacted with tRNA was found to have had generated mRNAs and genetic apparatus. Kin sociogenesis based on altruistic behavioral network could have had generated a hierarchical queen-worker-type society capable of working as a learning neural-net which could make the kin society a more adaptive machine or upper-level individual (super-organism). This logic seems to underlie in the emergences of genetic apparatus (intracellular hierarchical tRNA ribo-organism society), bee superorganism, and an animal body machine (hierarchical germ-line-somatic-line diploid-cell society). General semeiogenesis in these societies was discussed from de Saussuire's viwpoint[13] of arbitrary correspondences between *signifian and signifié*.

Key words: tRNA, poly-tRNA theory, origin of genetic code, sociogenesis, biomachine, semeiogenesis, hierarchical sociogenesis, neural net, altruistic behaviour

1. Introduction : Living organisms are well-made machines (Dawkins[9]), but who were or are machine-makers has long been an important unsolved problem. In this report, the answer to this question will be considered from the aspects of hierarchical sociogenesis which could help semeiogenesis and machinogenesis as generalized cultures of the hierarchical society.

Recently, poly-tRNA theory (Ohnishi, Ohnishi et al. [1-7]; Hokari *et al.* [8]) has proposed that RNA transcripts from the *Bacillus subtilis trrnD* and *rrnB* operons could be relics of early peptide-synthesizing RNA machines. Both of these RNA transcripts have a common structure, (5') 23S rRNA-16S rRNA-5S rRNA-(tRNA)n (3'), where $(tRNA)_n$ denotes a tandem arrangement of tRNAs (plus short spacers) (n = 16 for *trrnD* operon, as shown in Fig. 1[A], and = 21 for *rrnB* operon as in Ohnishi[1,7]). These arrangements of n tRNAs are here called "poly-tRNA structure", which could be a tandem association of tRNA-riboorganisms.

2. Poly-tRNA model and the emergence of mRNAs and genetic codes: Hypothetical peptides having amino acid (aa) sequences (seq.'s) identical to the ordering of the aa specificities of tRNAs in *trrnD- and rrnB-* poly-tRNAs are called "*trrnD-* peptide" ("NSEVMDFTY W HQGCLL" in Fig.1[B]) and "*rrnB-* peptide" ("VTKLGLRP AMISMDFHGINSE"), respectively. Aa seq. similarity searches from protein-seq. databases (PIR, SWISSPROT) revealed (1) that the 16-aa *trrnD*-peptide shows an 11 aa-match to the *Salmonella typhyrium* phosphoglycerate transporter protein B, and a 7 aa-match to the *E. coli* glycyl-tRNA synthetase alpha subunit (Figs.1, 2), and (2) that the 21-aa *rrnB-* peptide shows a 12-aa match to the *Saccharomyces cerevisiae* glyceraldehyde-3-phosphate dehydrogenase[1,7]. DNA sequceces encoding these aa-seq. segments were further found (1) to be significantly complementary to the tandem arrangement of 16 or 21 tRNA-anticodons in these poly-tRNAs, and further to be homologous to $tRNA^{Gly}$ (in *trrnD*) (Figs.1, 2; Ohnishi[3,5]) or $tRNA^{His}$ (in *rrnB*) [7], respectively. The poly-tRNA region ($tRNA^{His}$-Gln-Gly-Cys-Leu-Leu region) in the *trrnD* operon was further concluded to be most plausibly homologous to the DNA sequences encoding the *E. coli* GlyRS α subunit aa residues 85-303 and the *Synechococcus sp.* F_0-ATPase a subunit aa residues 1 -170, as is shown in Fig. 4. Further analyses revealed that various oligo-tRNA segments in these poly-tRNAs are significantly homolo-gous to various house-keeping mRNAs[1-5], M1 RNA[6], and 16S rRNA[8] (Fig.3). Accordingly, some early tRNA had interacted with anticodon regions of poly-tRNAs (on primitive ribosome) (See the poly-tRNA model in Fig.1), which enhan-ced selection of base-replacement mutations generating complementarity between presumptive codons in presumptive mRNAs (which were $tRNA^{Gly}$ or $tRNA^{His}$) and anticodons in poly-tRNAs. After the establishment of these base-complementarity, tRNAs and mRNAs evolved as modern elements of protein-synthesizing machinery.

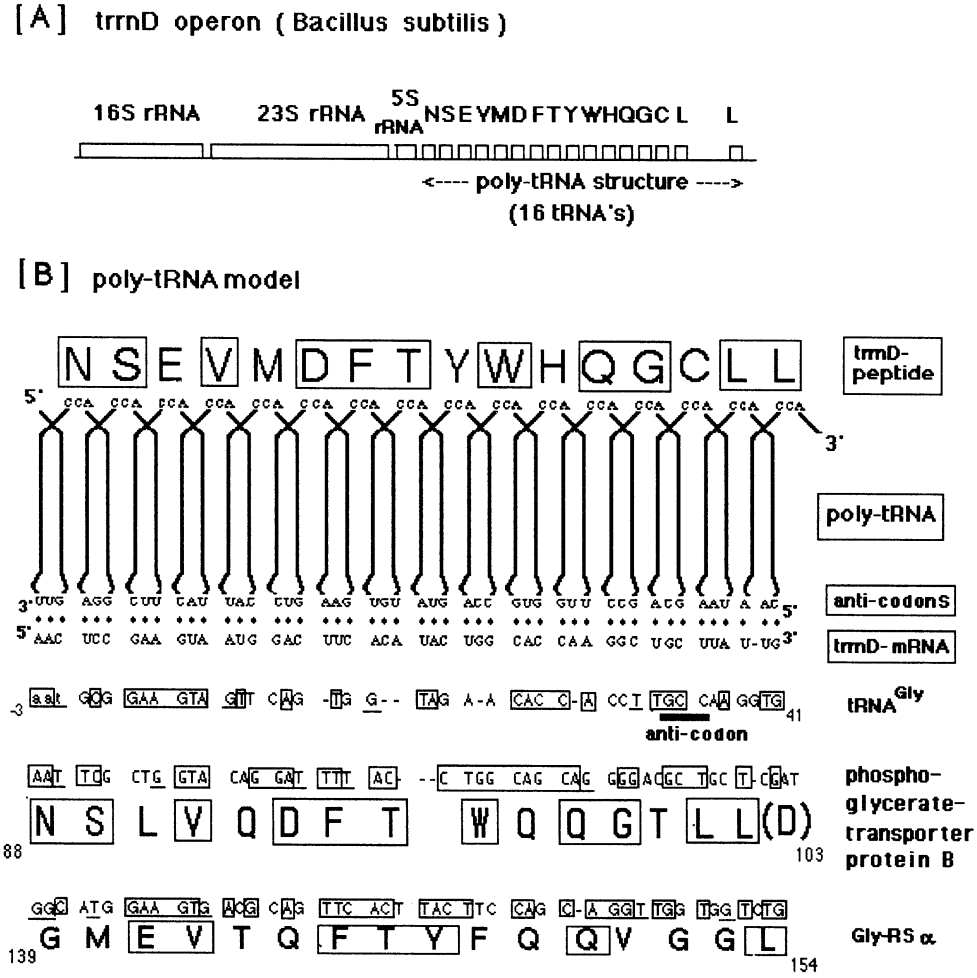


Fig. 1. A poly-tRNA model for early peptide-synthesis and the emergence of a *trrnD*- type primitive mRNA from $tRNA^{Gly}$. poly-tRNA = poly-tRNA region of the RNA transcript from the *B. subbtilis trrnD*- operon. *rrnD*- peptide = a hypothetical 16-aminno acid-peptide whose amino asid seuence is the same order of the 16 amino acid specificities of the 16 tRNAs in the poly-tRNA. *trrnD*- mRNA = a hypothetical early 48-base-mRNA complementary to the 16 anti-codons of 16 tRNAs in the *B. subtilis trrnD* poly-tRNA. $tRNA^{Gly}$ = $tRNA^{Gly}$ gene in the *Bacillus subtilis trrnD* operon. phosphoglycerate-transporter protein B = pgtB gene in *Salmonella typhimurium*. Gly-RS = Glycyl-tRNA synthetase alpha subunit (*GlyS* gene) in *E. coli*.

3. Hierarchical sociogenesis as evolutionary bio-machinogenesis : Contemporary genetic apparatus can be therefore considered as a result of hierarchical sociogenesis of intracellular kin tRNA-riboorganisms, which had generated queen-tRNA organisms (modern tRNAs) and worker tRNA-organisms (mRNAs, rRNAs, M1 RNA, deoxy-RNAs, etc.). Similar hierarchical sociogenesis seems to have generateed modern animal-body machines consisting of queen diploid unicell organisms (fertile germ-line cells) and worker unicell organisms (sterile somatic cells), and a bee-super-organism machine consisting of a queen-bee (fertile diploid female) and woker-bees (sterile diploid worker females). Thus there seems to be a general evolutionary logic generating the mechanism by which hierarchical sociogenesis could have evolved to become a well-made machine (Ohnishi[10]). RNAs could have emerged as a replicator ribo-organism within primitive unicell-organism, and earliest tRNAs can be considered as newly emerged tRNA riboorganisms which can live exclusively in intracellular environments. Co-operative and altruistic behaviours among kin tRNA-organisms could have thereafter generated hierarchical divisions of work in the kin tRNA society, probably by an evolutionary logic principally similar to Hamilton's[11] kin selection rule. Thus the hierachical tRNA society (comprizing queeen tRNAs and worker-tRNAs) have further evolved as a

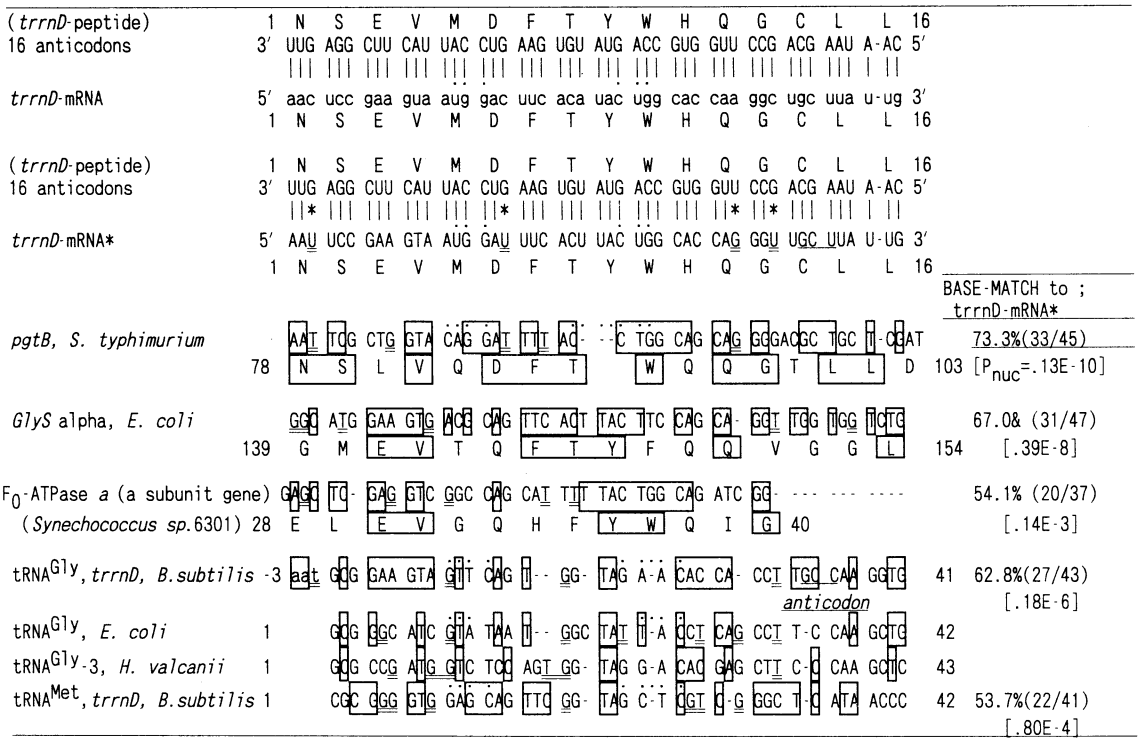


Fig. 2. Alignment of *trrnD* -mRNA with *pgtB* (*Salmonella typhimurium*) and *GlyS* (*E. coli*) genes. (Based on ref.3,5). Base- and amino acid- matches to *trrnD*-mRNA and *trrnD* -peptide are boxed. Base complementarities of Watson-Crick type and wobble type with 15 anticodons are indicated by "|" and "*", respectively. "XXXX...YYYY" denotes base-complementarity either in the D stem-forming regions of tRNAs, or in their corresponding regions of *pgtB* mRNA and *trrnD*-mRNA(*). Statistical evaluation of base-match level was made by computing $P_{nuc}(m,n) = \Sigma^{n}_{i=m} C_{n,i}(1/4)^{i}(3/4)^{n-1}$, (where $C_{n,i} = n!/[i!(n-i)!]$), which denotes matching probability by chance giving the observed m or more base-matches in the n-base alignment. [.xxE-xx] given below base-match level, %(m/n), denotes that $P_{nuc}(m,n) = .xx \times 10^{-xx}$. Base-match level(73.3%) between *pgtB* and *trrnD*-mRNA* gives $P_{nuc}(33,45) = 0.13 \times 10^{-10}$, confirming that *pgtB* DNA sequence is homologous to *trrnD*-mRNA*. The 62.8% match ($P_{nuc}=0.18 \times 10^{-6}$) between tRNA-Gly and *trrnD*-mRNA* suggests possible homology between these sequences, which is further confirmed in Fig. 4. Sequence data are from GenBank Databade and from Yang, YL, Goldrick, D., & Hong, JS. (J.Bacteriol. 170: 4299-4303, 1988.for *pgtB*.).

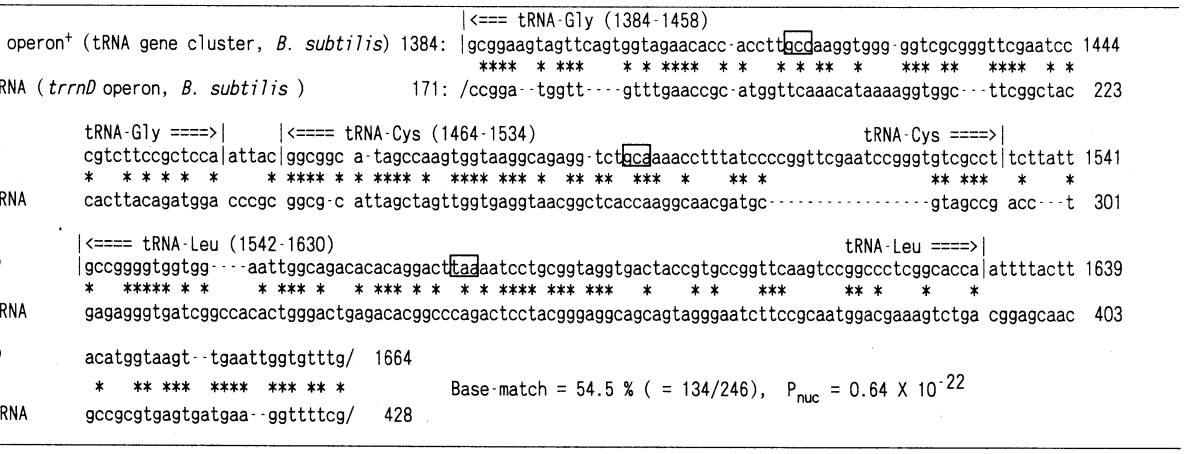


Fig. 3. *B. subtilis* 16S rRNA aligned against the poly-tRNA region ($tRNA^{Gly}$-$tRNA^{Cys}$-$tRNA^{Leu}$-spacer) of the *B. subtilis trrnD* operon. Aitiocodons are boxed. + Sequence data and base numbering is based on Wawrousek(1984). The resulting Base-match level and the P_{nuc} value given in this Table shows genuine homology of these two sequences, confirming that 16S rRNA had evolved (directly or indirectly) from a primitive *trrnD*-type poly-tRNA molecule. This sequence similarity was found by a dot-matrix method.

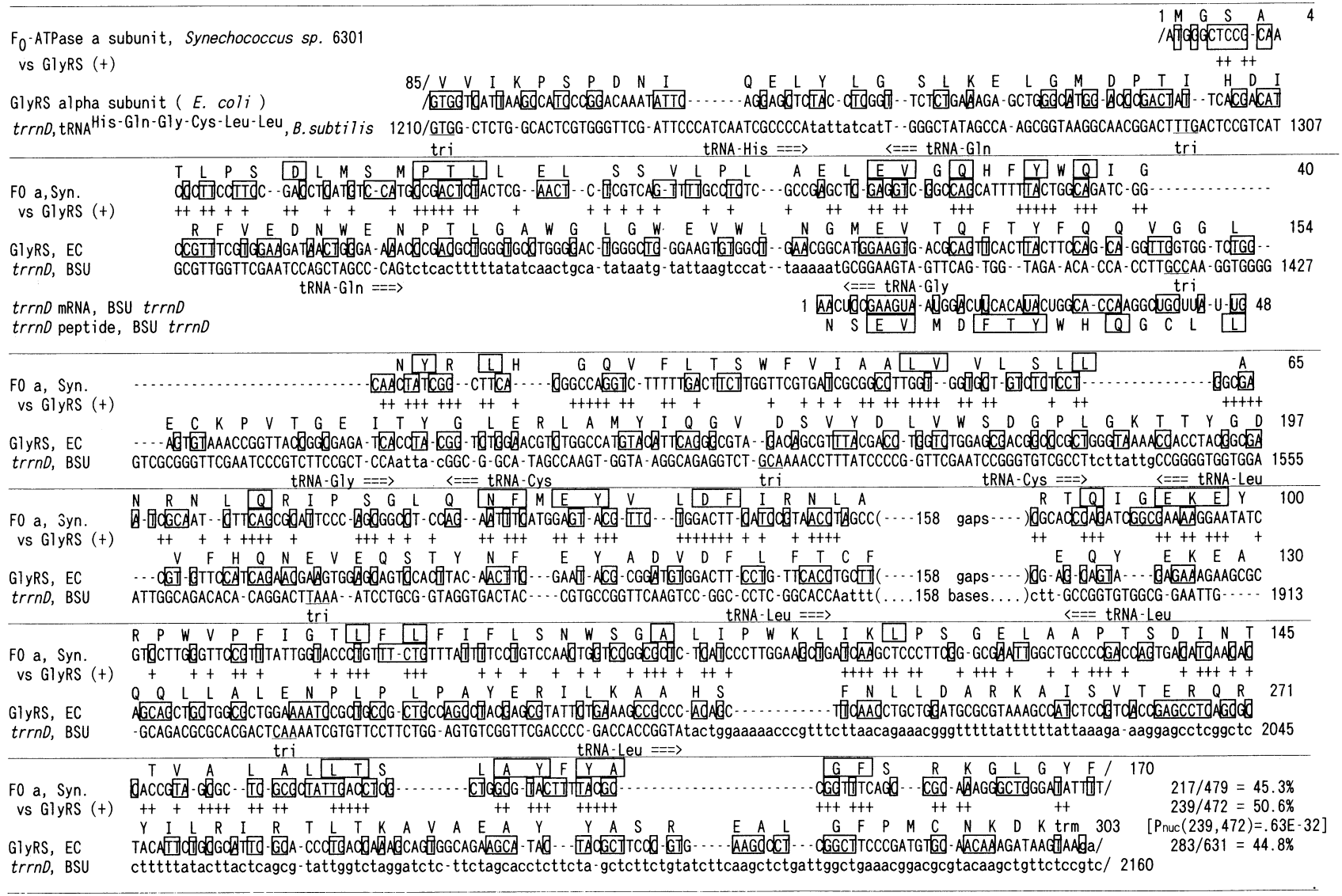


Fig. 4. Alignment of the *E. coli* (EC) glycyl-tRNA synthetase (GlyRS) alpha subunit gene (*GlyS*) and the *Synechococcus sp.* F_0-ATPase a subunit gene (*a*) against the poly-tRNA region (tRNA$^{His\text{-}Gln\text{-}Gly\text{-}Cys\text{-}Leu\text{-}Leu}$ region) from *Bacillus subtilis* (BSU) *trrnD* operon. Base-matches to the *trrnD* poly-tRNA region and deduced amino acid matches to GlyRS are boxed. Sequence data are from GenBank Database. Base and amino acid matches to *GlyS* /GlyRS are boxed, and/or indicated by "+".

machine for making proteins (as tools or machines in the tRNA-riboorganism society. Proteins are also inner tools of the cell-individual organism), and for transmitting genetic informations to the next generation. The above-mentioned selections of base-replacement mutations giving comple-mentarities bet-ween presumptive codons and presumptive anticodons can be therefore considered as a semeiogenetic cultural process (in tRNA society) towards generating "signe or semeiotic coding system". Semeiosis is a cultural phenomenon found in human and animal cultural society. Genetic codon-anticodon system clearly shows a characteristic features (such as arbitrariness, etc. See Shibatani[11],Ohnishi[4]) of signe or semeiotic system[13], and the question why such typical signe or semeiotic system can be found within a cell seems to be reasonably solved by answering that genetic codon-anticodon system is a typical semeiotic phenomenon evolved as a (generalized) culture or a meme of intracellular kin tRNA-society. Why could hierarchical sociogenesis make the kin-society a well-made machine in every case of genetic apparatus, animal-body machine, and bee super-organism ? A most plausible logic would be that hiearchical kin-society could function as a kind of "learning neural-net machine"[14] capable of learning inner and outer input informations and re-building the society as a more adaptive machine probably based on using the final output from the queen to the next generation . It is important to clearify real informations (inputs and the final output) of the hierarchical society machine. Some feed-back of information could be made by queen-to-worker interaction in these kin society as is known in hymenopteran and termite eusociety (Wilson[15], Sakagami[16]). Brain is also a kin-society of unicell nerve-cell worker-individuals , wich is a sub-society or a part of somatic-cell subsociety of the hierarchical kin-society of unicell-organisms making an animal-body super-organism (of unicell-organism) machine.

4. Origin of cognitive system, thinking system, and generalized semeiotic system : Typical signe or semeiotic systems such as genetic triplet codon-anticodon system, bee dance-language system, immuno-recognition system, and human language system can be found in tRNA-society (= super-riboorganism)[1-8], bee-society (= super-organism)[17,18], immune-cell society (= a subsociety of hierachical germ line-somatic line diploid-cell society)[19], and human society[13], respectively. What is or are the essential feature(s) common to these four very different semeiotic systems ? In the human language system, semeiosis is considered to be a process to correspond *signifian* (= signifier) (which ia intra-brain acoustic image of uttered word) to *signifié* (= the signified) (which is intra-brain concept whi can carry "meaning") (de Sausseure[12]). In a simple neural system capable of making habituation and sensitization in *Aplasia* , stimulating the siphon excites the sensory neuron which in turn stimulates the motor neuron in the gill, both directly and via the interneuron (Rowe[14]). In this system[14] and the bee-super-organism system[17,18], a sensory neuron (= a worker-cell in animal body) or a sensory bee (= a worker-bee) accepts and carries outer information and express specific signe or bee dance-language, which is in turn transferred to a motor neuron (another-worker cell) or motor bee (which is another worker-bee), respectively. These two systems closely resemble with each other, in both aspects of function and sociogenesis. Possible logical counterparts of *signifian* and *signifié* are summarized in Table 1. Arbitrary correspondence between *signifian* and *ignifié* is a most characteristic feature in mature semeiotic system (de Sausseure [13]; Shibatani[12]; Ohnishi[4]). Such arbitrary correspondences seem to be generally found in these four semeiotic systems, as shown in Table 1.

Accordingly, cognitive and/or thinking system(s) seem to be special case(s) of generalized cognitive systems which could essentially be some kind of hierarchical neural-net-type kin-society.

5. Conclusion: Based on poly-tRNA theory, genetic apparatus was concluded to have evolved by hiearchical sociogenesis of intracellular tRNA-ribo-organisms. Similar hierarchical sociogenesis generating society-machine can be found in the queen-worker bee-society (= superorganism) and in the animal-body machine (= hierarchical germ line-somatic-line kin society). A queen-worker-type (altruistic) behavioral network in a hierarchical kin society tends to evolve to be a biomachine by functioning as a kind of learning neural network. Genetic code-system, bee-dance language, and human language share similar semeiotic features such as arbitrary correspondence between Sausseure's *signifian* and *signifié*. Genetic codon-system is therfore a cultural phenomenon in tRNA society.

Acknowledgements: Partially supported by the Scientific Research Grant (#06302085) from the Japan Ministry of Education, Science and Culture. Discussions are acknowledged to Profs. Y. Hushimi, M. Gō, K.Nagano, A. Shibatani, and M. Dan-Sohkawa.

References

1. Ohnishi, K (1993a), Evolution from semi-tRNA to early

Table 1. Generalized semeiotic systems and arbitrary correspondences between signifier (*signifian*) and the signified (*signifié*). (See also Ohnishi[10] .)

Outer code or input information	signifier or *signifian* (Image)	the signified or *signifié* (Concept)	arbitrary correspondence between *signifian* & *signifié* is maintained by :
[I] Language system in human-society culture (and in brain) (de Saussure, 1989)[13]			
uttered word	acoustic image (produced by sensory neurons ?)	concept (or image-recognizing ability) (of some areas of brain ?)	social consensus (of human individuals)
[II] *Aplasia* simple nerve-system capable of making habituation and sensitization (Based on Rowe, 1994)[14]			
(stimulation to siphon)	stimulated state of a sensory neuron	"primitive concept ?" or image-recognizing ability of a motor neuron	?
[III] Dance language system in the culture of bee society (= super-organism) (Based on Griffin, 1984,1992)[17,18]			
(position of flower, etc.)	bee-dance (of a sensory bee)	dance-recognizing ability (or concept ?) (of a motor bee)	social consensus (culture or meme in the bee society)
[IV] Immuno-recognition system in immune-cell (sub-)society (Based on Klein and Hořejši, 1997)[19]			
(non-self epitope of antigen)	peptide(epitope)-MHC class II antigen complex (presented by and on antigen-presenting cell)	epitope-specific antibody-producing ability (of B lymphocyte)	genetic and educated (cultural) consensus in immune cell-society (after clonal selection)
[V] Protein-encoding system in the generalized culture of tRNA-riboorganism society			
codons on mRNA	anti-codons (sensory organ) of tRNA	amino-acid-binding specificity of tRNA	aminoacyl-tRNA synthetase as a cultural product of tRNA-society

protein-synthesizing RNA molecule. In: Satoh, S. et al. (ed.), *Endocytobiology V* , 407-414,Tubingen Univ. Press.

2. Ohnishi, K (1993b) Molecular evolution of ion channels in central nervous system. *Ann. New York Acad. Sci.* 707: 524-528.

3. Ohnishi, K (1993c): Poly-tRNA theory on the origin and evolution of mRNA and genetic codes. In: Takagi,T et al.(eds.), *Genome Informatics IV*, pp.325-331, University Academy Press, Tokyo.

4. Ohnishi, K.(1995): Non-paradoxical origin of the genetic codes. I, II. (in Japanese). *Seibutsu Kagahu (Biolobical Science)*, 47 (1): 10-23 ; 47(3): 155-166.

5. Ohnishi,K(1995), Poly-tRNA structures as early protein-synthesizing RNA apparatus : Evolution from proto-tRNA to most primitive mRNA's. *Viva Origino* 23(3), 237-254.

6. Ohnishi, K., Suzuki, T., Suwa, K., Yanagawa, H. (1998a), The origin of ribonuclease P RNA (M1 RNA), as viwed from Poly-tRNA theory.*Viva Origino* 26: 251-258.

7. Ohnishi, K, Tanaka, H, Yanagawa, H(1998b): The origin of DNA-binding domains, as viewed from poly-tRNA theory. *Nucleic Acids Symp. Ser. (Oxford)*, 39: 251-252.

8. Hokari, S, Ohnishi,K (1998), The origin of 16S rRNA. *Nucl.Acids Symp. Ser. (Oxford)*, 39: 159-160.

9. Dawkins, R (1986), *The Blind Watchmaker.* Harlow Longman Scientific & Technical, London.

10. Ohnishi, K(1990), Evolutionary meanings of the primary and secondary structures of "ur-RNA". In: Gruber B & Yopp, JH (eds.), *Symmetries in Science IV*, pp. 147 - 176, Plenum, New York.

11. Hamilton, WD (1964), The general evolution of social behaviour. *J. Theor. Biol.* 7: 1-52.

12. Shibatani, A (1987): On structural biology. *Revista di Biologia --- Biology Forum*, 80: 558-564.

13. de Sausseure, F. (1989): *Cours de linguistique générale. Edition critique par R. Enger.* Harrasowitz, Wiesbaden.

14. Rowe, G.W.(1994): *Theoretical Models in Biology. The Origin of Life, the Immune System, and the Brain.* Clarendon Press, Oxford.

15. Wilson, EO (1975) : *Sociobiology: The New Synthesis.* Belnap Press, Cambridge, Mass.

16. Sakagami, SF, Maeta, Y.(1987): Sociality, induced and/or natural, in the basically solitary small carpenter bees (*Ceratina*). In: Ito, Y et al.(eds.), *Animal Societies. Theories and Facts.* Japan Scientific Societes Press, Tokyo.

17. Griffin,DR (1984), *Animal Thinking.* Harvard U.Press.

18. Griffin,DR (1992),*Animal Minds.* U. of Chicago Press.

19. Klein, J, Hořejši, V (1997), *Immunology, 2nd ed.*, Blackwell Science Ltd., Oxford.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Long-Term Increase of Complexity and Functional Diversification by Contingent Mutations in Computational Algorithms

Satoshi OHASHI*, Shinichiro YOSHII, and Yukinori KAKAZU***

*Complex Systems Engineering, Hokkaido University
N-13, W-8, Kita-ku, Sapporo 060, Japan.
**Department of Computer Science, University of Liverpool
Liverpool L69 7ZF, UK

Abstract

In simulating a complex adaptive system on a computer, an agent can be regarded as an entity in order to process information. In this paper, therefore, we adopt a general computational model, that is, a Turing Machine (TM), to realize our agent. Any algorithm can be expressed as a TM or a readable tape that can be emulated when a Universal Turing Machine reads that tape. We have developed an abstract complex adaptive system called PROTEAN as our simulation model. This paper explores how long-term complexity in computational algorithms is affected by contingent mutations with an artificial life model. Unlike many researches that have investigated a relation between a macroscopic system dynamics and a mutation rate, we, at the same time, focus on functional diversification of agents from the viewpoint of an internalist perspective.

key words: simulation model, evolution,
long-term behavior, complexity

1 Introduction

This paper explores how long-term complexity in computational algorithms is affected by contingent mutations with an artificial life model. Unlike many other studies, which have investigated the relationship between the dynamics of macroscopic systems and mutation rates, we also focused on the functional diversification of agents from an internalist perspective.

We constructed a simulation model called the PROTEAN which is an ecosystem consisting of computer programs. With this model, interaction takes place independent of the location of the ecological resource, unlike with Avida[1]. Unlike Tierra[3], there are no explicit parameters to control the dynamics, such as a parameter to reap constituent machines in the system. The sole condition for a program to survive is its being able to accept its local resource more effectively than the ohers, and self-reproduce, where possible. The fitness of each program is dependent dynamically on its relation to the others within the system and the consequent global environment thereof.

In previous research, the short-term behavior of the programs was characterized by self-organization of the interaction network and self-reproduction. All the programs ran deterministic calculations. Therefore, once a stable ecosystem had been constructed, no more innovation occurred. In this paper, we introduced a mutative operation to the calculation result, thereby, not affecting the calculation process itself. Here, the programs achieved competitive coexistence within the ecosystem. We were able to observe the long-term behavior of the programs, and found that the algorithms were able to adapt to a noisy environment. Computational complexity and functional diversification were found to be affected by contingency.

Section 2 describes an ecological model of computer

Figure 1: Ecological Resource



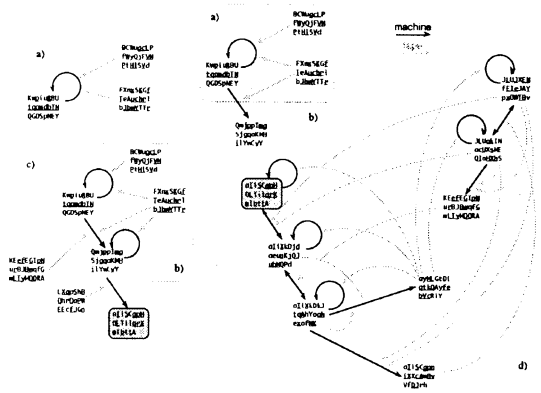
programs. Section 3 shows the characteristics of this model. Section 4 simulates the long-term behavior of the computer programs. Lastly section 5 discusses the simulation results, and then concludes this paper.

2 PROTEAN Simulation Model

In simulating a complex adaptive system on a computer, an agent can be regarded as an entity in order to process information. In this paper, therefore, we adopt a general computational model, that is, a Turing Machine (TM), to realize our agent. Any algorithm can be expressed as a TM or a readable tape that can be emulated when a Universal Turing Machine reads that tape. We have developed an abstract complex adaptive system called PROTEAN as our simulation model. PROTEAN stands for *Platform on Recursive Ontogenetic Turing machine Ecosystem for Autopoietic Networks.*

Figure 1 shows this ecosystem which consists of description tapes of TMs. Its total resource is a static volume of memory. Each TM occupies its own memory block, the size of which depends on its description length. This model consists of a large number of TMs that can be regarded as agents in a computationally universal ecosystem. The TMs interact with each other by means of reading a tape (i.e. which encodes a TM) chosen at random from within the ecosystem. Driven by the combination of its algorithm and the reading of tapes, each TM continues to calculate step by step, until all calculation has finished. The TMs interact, in parallel, based on their own instructions, and output the result

Figure 2: Emergent highly complex set of connections between TM agents.



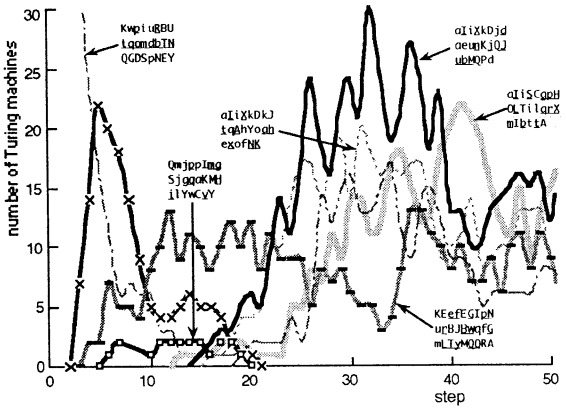
of their interactions as new tapes which will dictate new algorithms (TM). Each tape becomes a new member of the ecosystem and accordingly occupies a new memory block. Sometimes, an existing tape will be eliminated and the calculation terminated in order to make a new memory at this point. Therefore, this model has no survival time or reaper as seen in Tierra.

3 Simulations

In our previous research [4] [5], we showed how self-reproductive networks could be achieved from a population of different species of TM. Figure 2 illustrates the reaction pathways between the TMs. Figures 2 a), b) and c) denote the growth of interaction networks, where the initial TMs generated new TMs, one after another, through their interactions. Figure 2 d) shows the final reaction network that emerged in the ecosystem. Whatever the initial TMs may be, in the final version the network will usually consist of a small number of different TMs to the original. In the growth of this reaction network, various TMs appeared and disappeared. Figure 3 shows the population dynamics achieved through the emergence new TMs. Up to about the 20th step, we see a growth network. After that, once a stable reaction network had been achieved, the members of the network remained fixed.

To understand the behavior of these TMs, we

Figure 3: Population Dynamics through Emergence of New TMs.



measured some characteristic variables, that is, time complexity and space complexity within each interaction between TMs. According to one definition [2], the complexity of a TM is characterized by the following properties: time complexity, which means the amount of time required to perform a computation, which will correspond to the steps of a head moving in the case of a TM; and space complexity, which, on the other hand, refers to the amount of storage space required by a computation. The space complexity of a TM is defined as the number of tape cells required. We observed that the self-organizational dynamics would reach a certain boundary depending on the level of complexity involved. We found that they would not diverge ad infinitum, even though the halting problem means that the termination of the calculation process cannot be predicted. Figure 4 shows the transition of correlations between both complexities, with respect to the average number of constituent TMs in the above simulation. These figures indicate that the average complexity of constituent TMs increases and self-organizes to a certain attractor.

4 Introduction of Mutation and Simulations

The mutations employed will occur contingent upon the emergence of new tapes. This means that turbulence is not introduced directly into operations

Figure 4: Self-organized complexity of TM agents.

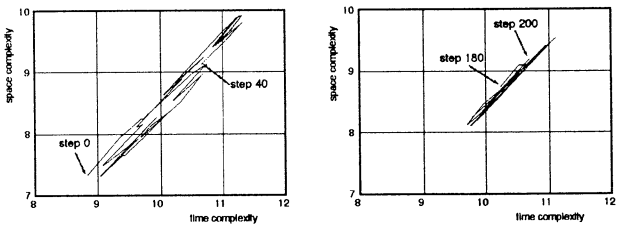
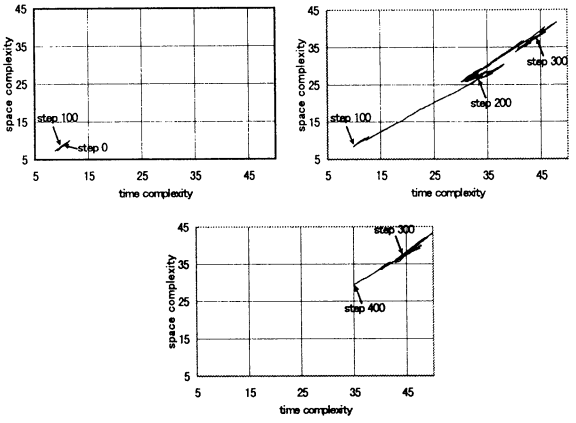


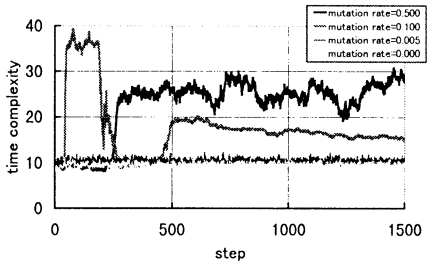
Figure 5: Transition of Complexities



between TMs. Instead, it is introduced into the environment. The simulation results show that such a noisy environment can't result in a situation where only one, or certain species of TM can engage in a stable relation of competitive coexistence. This indicates that a noisy environment is clearly different from the case where there are no mutations. Figure 5 shows the transition of the correlation of complexities in the case where mutations are employed. These complexities increase and decrease successively, because the stable reaction network is jammed by mutated TMs attempting to construct new networks. Figure 6 shows the time series of the time complexities in the case of various mutation rates.

The setting of the mutation rate was exaggerated as it showed little variation which, therefore, did not affect the dynamics dramatically, except in the case where there was no mutation. The complexities show some large shifts and long-term small oscillations. This oscillation means that there is a stable core network whose members show no change or only change partially. In the case of a low mutation rate, new TMs

Figure 6: Long-term Fluctuation of Complexity



generated by mutation were eliminated through interaction with stable network members such as in Figure 2 d). After 500 steps on the middle of mutation rate, the level of complexity gradually reduced. Each TMs' algorithm optimizes to faster self-reproduction with keeping the stable network.

When the mutation rate is elevated, the complexity of interactions increases compared to with a low mutation rate. Lower mutation rate allows the system to have many kinds of offspring algorithms of lower complexity. It follows that such agents, whose complexity is lower, have an advantage in surviving in the system. As a result, the system is dominated by agents who can effectively self-reproduce. On the other hand, in the case of a higher mutation rate, agents of higher complexity will survive because they are able to utilize other tapes whose complexity is also higher. Therefore, a highly complex set of connections involving a more competitive collaboration is observed. This is because agents with a higher complexity have the advantage of being able to accept other tapes due to their redundant functionality. Such agents appear to be robust to a dynamically changing environment.

5 Discussion and Conclusion

This section discusses the obtained simulation results from the viewpoint of the evolutionary dynamics of computer programs. PROTEAN has the ability to describe and produce any kind of computer program in the form of TMs. These TMs interact with each other, and generate new algorithms in turn. In this process, once a self-reproductive network has been achieved, that network is dominating and self-organizing. This is a natural dynamics, as the network is maintained even though there is a low rate of mutation. With a higher rate of mutation, the network collapses, the TMs optimize reproduction and adapt to surrounding resources.

The evolutionary scenario of the computer programs in PROTEAN was as described below. In the early phase of the ecosystem, the diversity of TMs increased rapidly in a chain reaction. These TMs were able to achieve a self-reproduction network, or self-organize to a particular attractor. Further, on introducing mutative operations, the TMs ware forced into a competitive coexistence. The TMs adapted their function to a varying environment. Basically, it can survive as the TMs are able to use the resources more effectively. However, when various TMs continued to exist in an ecosystem with a noisy environment, they acquired redundancy through interaction with various species. In future work, we will study the evolutionary dynamics of PROTEAN in even greater depth.

References

[1]Adami, C. and Brown, T. 1994. Evolutionary Learning in the 2D Artificial Life Ssytem 'Avida'. Artificial Life IV. The MIT Press. 377-381

[2]Brookshear, J. G. 1989. Theory of Computation: Formal Languages, Automata, and Complexity. The Benjamin/Cummings Publishing.

[3]Ray, T. S. 1991. An Approach to the Synthesis of Life. Artificial Life II Proc. Vol. X. Addison-Wesley. 371-408

[4]Yoshii, S., Ohashi, S., and Kakazu, Y. 1998. Self-organized Complexity in Computer Program Ecosystem, Artificial Life VI, pp. 483-488, The MIT Press, 1998.

[5]Yoshii, S., Ohashi, S., and Kakazu, Y. (1998b). Modeling of Emergent Ecology for Simulating Adaptive Behavior of Universal Computer Programs, From Animals to Animats 5, Proceedings of the Fifth International Conference on Simulation of Adaptive Behavior, The MIT Press, 1998.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Team Plays of Soccer Agents based on Evolutionary Dynamic Formations

Tetsuya MURATA Masahito YAMAMOTO Keiji SUZUKI Azuma OHUCHI

† Research Group of Complex Systems Engineering,
Graduated School of Engineering, Hokkaido University,
Kita 13 Nishi 8, Sapporo, Japan
{tetuya, masahito, suzuki, ohuchi}@complex.eng.hokudai.ac.jp

Abstract

In this paper, we propose the dynamic formation of the soccer agents for realizing the autonomous team play. The formations can be thought as the arrangements of the positions assigned to the players as home positions to proceed the specific roles in the team plays. Thus, each player basically moves around the neighbor of the assigned home positions and plays to realize the effective team play.

To search the appropriate home positions in the Soccer Server domain, we adopt the genetic algorithms as robust optimization methods. However, it is very difficult problem to judge the effectiveness of the position arrangements. Because even if the team succeeds to make the effective position arrangement, the team can't always win the games. Namely, the result of the games includes many noises in every time.

So, in order to evaluate the position arrangements, we suggest the behavior based evaluation function which consists of the frequency of each agent's behavior, i.e., shoot, pass, search ball and etc.

We will show the experimental results applied the proposed methods on the soccer server.

1 Introduction

Recently, as a test-bed for multi-agent systems, Soccer Server is served for the research of realizing the cooperative team behavior to defeat opponent teams in complex game environments. In the soccer game problem, it is very difficult to connect the behavior selection problem in each agent with the problem of realizing the effective autonomous team play. In this paper, the important factor for realizing the autonomous team play, we assume, is to make the appropriate formation of the players. In addition, the formation should dynamically change in accordance with the situation in the game. Here, the formations can be thought as the arrangements of player's home positions in the soccer field. Therefore, we prepare the agents embedded with the behavior selection based on a given home position. Then we try to search the adaptive position arrangements of the agents, which will be dynamically changed, for realizing the autonomous team play.

While, the result of the games includes many noises in every time. Namely even if the team having the effective position arrangement, the team can't always win the games. So, it is required that the robust search method and the evaluation method to make the appropriate formations in the noisy environment.

As robust search method, we adopt the genetic algorithms with the improved genetic operations. As evaluation method that doesn't only depend the score of the games, we introduce the behavior based evaluation function. The behavior based evaluation function consists of the frequency of each agent's behavior in the given position, e.g., shoot, pass, search ball, and etc. Because if an effective team play can be realized, the agents in the team will become to play that the effective behavior related to win the team, such as, shoot, kick and pass, will be increased against to the time consuming behaviors and the ineffective behaviors like as searching the loose boll.

2 Dynamic Position Arrangements to realize Dynamic Formations

We use the analogy of the adaptive formation in real soccer games for constructing the autonomous team play with soccer agents. In real soccer games, players adaptively change the home positions according to the current situations. Thus, to realize the adaptability concerning the formations, the dynamic position

arrangement problem is proposed.

In this chapter, the proposed method will be explained.

2.1 Agents

To implement the dynamic position arrangements problem, each agent is required the fundamental activity such as catching a ball. So we construct the action-selection architecture of the agents based on the winner's architecture of Pre-RoboCup 96[Ogawara 96].

In the architecture, the agents can perform the several behaviors around one fixed home position. These behaviors are pass, shoot, search the ball, approach the ball, and go back own home position.

The selection of the behavior is done according to the distances between own home positions, the ball, and other agents. For instance, each agent approaches the ball if the ball is closed to itself, but it goes back own home position if the ball is far from itself.

2.2 Dynamic Position Arrangements

To adopt the proposed method, we improve the above architecture as the agents can take the several home positions in each game and can dynamically change current home position in accordance with the situation in the game. We describe the method as follow:

- We prepare the candidates for the home positions. That is, the field is divided into $m \times n$ square areas.Then the centers of each square area apply to the candidates assigned to the agents as own home positions. Each candidate is indexed from 0 to $m \times n - 1$.
- Each player can use H home positions. These are assigned from $m \times n$ candidates. In this case, it is allowed to assign the same home position in H positions.
- To introduce the condition of changing current home position, the agent architecture is improved. That is, the agents should select the current home position with the minimal distance to the ball from the assigned home positions.

The agents with the improved architecture can move around the home positions. Therefore the agents will be expected to perform the suitable behavior according to the current situations if they are assigned the appropriate home positions.

2.2.1 Evaluate the Formation

To search the appropriate home positions, the evaluation function for the arrangements has to be constructed. However, it is very difficult problem to judge the effectiveness of the position arrangements. For example, the evaluation function can't be constructed with only the score of games since the score includes several noises. Therefore, we propose the behavior based evaluation function as follows;

In the team level, we assume if the total occurrence of the pass behavior and the shoot behavior in the game are increased than other behavior, the position arrangements may be effective even if these behaviors didn't succeed.(It requires the notice that the commands of behaviors are often failed with the noise in the Soccer Server.)

In the individual level, each agent should dash to catch the ball immediately when the ball is closed to the agent. While, the agent should go back own home position when the ball is far from the agent. In addition, each agent shouldn't lose the ball position because the search of the lost ball requires the time consuming behavior and the activities of the agents quietly decreased.

Thus, the number of the occurrence concerning with these behaviors is considered to evaluate the position arrangements. Of course, the number of the team getting the goal is added to evaluate the arrange of positions. Thus, we suggest the behavior based evaluation function which consists of the frequency of each agent's behavior and the score. The constructed evaluation function is described as follow:

$$fitness = (\sum pass/10) \times (\sum shoot/2) \times \frac{\overline{predict}/3}{(\overline{go_home}/20) \times (\overline{search}/50)} \times (\frac{Goals}{10} + 1) \quad (1)$$

where

$\sum pass$: the total of the frequency of pass in the team

$\sum shoot$: the total of the frequency of shoot in the team

$\overline{predict}$: the average of the frequency each player approached the ball

$\overline{go_home}$: the average of the frequency each player went back own home position

$\overline{search}$: the average of the frequency each player searched the ball

$Goals$: the frequency of goals the team got

2.3 Searching Positions with GA

To search the adaptive home position arrangements against to an opponent team, a robust optimization method is required in this domain. Therefore, we use the Genetic Algorithm (GA,[Agui 93]) with inheritance of the fitness for increasing the robustness in noisy optimization space.

2.3.1 Coding

One chromosome express one formation in the game. Genotype consists of the indices of home position candidates. In case of N players and H home positions assigned to one player, the length of a chromosome is $N \times H$.

2.3.2 Inheritance of Fitness

As the improvements of GA, it is introduced the inheritance of the fitness. The fitness values of the parents are inherited to the fitness of offspring according to the resemble between the parent and the offspring. The inheritance of the fitness will be expected to decrease the effect of noisy search space. The fitness of the offspring is calculated by fit_{cal}, which is evaluation value of the above function according to the formation assigned by the offspring, adding the value, (fit_{old}), which is calculated based on the resemblance between the offspring and their parents. The resemblance is defined based on the ratio of inherited genes from the parent in the crossover operation.

3 Experiment

To confirm the effectiveness of the proposed method, we carry out the experiment. In each game for evaluating the formation of the chromosome, the winner's team of Pre-RoboCup 96 ([Ogawara 96]) is used as the opponent team. The opponent team already has the hand-coded fixed home positions. The arrangement of the opponent's positions can be thought as enough to perform the autonomous team play. Basically, the ability of the agents in both teams is same except for our agents being available the multiple home positions. Thus, we expect that the GA will find the dynamic position arrangements of the agents to perform the flexibility and the adaptability against the fixed position arrangement of the opponent team.

The detailed settings of the experiment are represented as follow:

- The number of players in the game : 11 vs 11
- The field is divided in 5 × 7 square area.
- 2 or 3 home positions are assigned to each player.
- The number of individuals : 30
- The number of generations : 30
- The crossover rate: 0.2
- The mutation rate: 0.1

3.1 Result

We show the formations with maximun fitness of last generation in which each player has three(A-team) and two(B-team) home positions in Fig.1 and Fig.2.

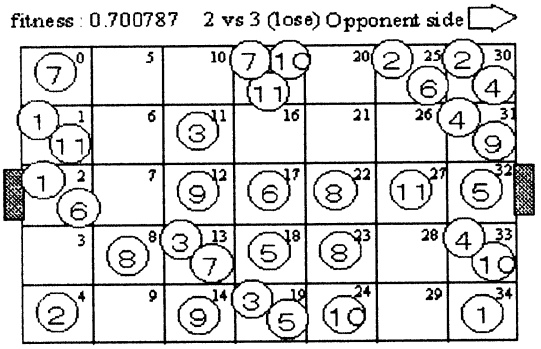


Figure 1: The Formation in which each player has 3 home positions of the maximum fitness

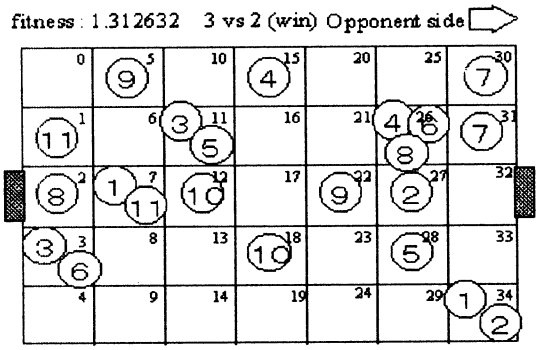


Figure 2: The Formation in which each player has 2 home positions of the maximum fitness

We show the results of games between the team with this formation and the winner's team ([Ogawara 96]) are in Table 1.

A-team(3 home positions)					
result	×	×	×	×	×
score	3 vs 9	1 vs 9	1 vs 7	0 vs 6	1 vs 11
B-team(2 home positions)					
result	×	×	△	×	○
score	3 vs 6	3 vs 4	4 vs 3	3 vs 6	4 vs 4

Table 1: The result of the games with the formation in which each player has home positions of the maximum fitness

And the frequency of home positions which each player selected in the match are in Table 3 and 4.

player home position number	player 1 1 5 2			player 2 25 30 4		
share(%)	1	42	57	37	4	59
player 3 11 19 13	player 4 30 31 33			player 5 18 19 33		
34	22	44	85	14	1	84	15	1
player 6 25 2 17	player 7 15 0 13			player 8 22 23 8		
29	21	50	92	7	1	65	34	1
player 9 31 14 12	player 10 33 15 24			player 11 15 1 27		
49	22	29	34	62	4	26	43	31

Figure 3: the frequency of 3 home positions which each player selected

player home position number	player 1 7 34		player 2 34 27	
share(%)	84	16	0	100
player 3 3 11	player 4 26 15		player 5 28 11	
27	73	23	77	21	79
player 6 3 26	player 7 31 30		player 8 26 2	
55	45	94	6	43	57
player 9 22 5	player 10 18 12		player 11 14 7	
61	39	44	56	36	64

Figure 4: the frequency of 2 home positions which each player selected

3.2 Discussion

As the Table 1 indicates, the result of A-team are better than B-team's. In order to analysis the result, we tried to search the frequencies which each player use each home position in the match(Table 3 and 4).

In these tables, the frequency which players in A-team use home positions are inclined. For example, though player 5 has one home position near opponent goal, this position is used very little. For this player, this position is useless. If this player didn't have this position, it could stay around the center circle and rule the midfield as midfielder(midfielder is very important position in soccer!).

In B-team, there are few players who must move the long distance between two home positions. For, each player can cover the field effectivity.

In addition, the search space of A-team's experiment is very wide. So, at first we should search by using B-team, then add one home position each player to defeet the champion team.

4 Conclusion

In this paper, we proposed the dynamic position arrangements of the soccer agents for realizing the autonomous team play. So we adopted the genetic algorithms for obtaining the adaptive position arrangements to defeat the opponent teams, and showed about the result of our experiment.

In our experiment, we found the adoptive position arrangements to defeat the winner's team of Pre-RoboCup 96 with changing current formation according to the situation in the game dynamically.

References

[Ogawara 96] Koichi Ogawara's program : University of Tokyo, Japan.(Winner of preRoboCup96) ftp://ci.etl.go.jp/pub/soccer/client/ogalets.tar.gz

[Agui 93] Takeshi,A., and Tomoharu, N.: 1993, "Genetic Algorithm"

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

An Analysis of DNA-based Computing Process

Takuo Hirayama†*, Toshikazu Shiba‡,
Masahito Yamamoto†, Kaori Tsutsumi‡, Shigeharu Takiya§,
Masanobu Munekata‡, Keiji Suzuki† and Azuma Ohuchi†
Research Group of Complex Systems Engineering
Graduate School of Engineering, Hokkaido University
Kita 13, Nishi 8, Kita-ku, Sapporo, 060-8628 JAPAN

Abstract

In this paper, we propose a new alternative method for solving a directed Hamiltonian path problem (HPP) based on Adleman's DNA-based computing paradigm to analyze some problems on DNA-based computing process. The applied method make us enable us to analyze the DNA sequences of generated paths and these behaviors in computing process. This method used E.coli for obtaining large number of one kind of DNA in order to analyze the DNA sequences. From the results of experiments, it was investigated that the distribution of generated paths in generating process. It showed some biological operations can occur noised which may make errors in search process. Some errors in generate process, which generate theoretically inexistent paths in the graph, were observed. Furthermore, we generated a HPP with 8 vertices and applied our method to it, and discussed practical problems on DNA-based computing.

1 Introduction

Adleman has shown that the instance of directed Hamiltonian Path Problems (HPP) can be solved by biological experiments using DNA [1]. Adleman's experiment are based on *generate and search* paradigm. The *generate and search* paradigm on DNA-based computing was supposed by Adleman. That paradigm generate all candidate solutions in parallel at first, and choose the correct solution from them by using biological operations. An instance of HPPs is well known to be a NP-complete problem, i.e., it requires exponentially time to solve as the problem size increases. DNA-based computing could be expected to overcome that difficulty.

In this paper, in order to analyze some practical issues on DNA-based computing process, we propose a new alternative method based on Adleman's experiment for solving a HPP with 7 vertices. Our proposed method enable us to know DNA sequences of generated paths. Therefore, the distribution of generated paths in DNA-based computing process is investigated and it can be expected that some critical errors, which generate theoretically inextistent paths, are observed, and discuss these causes. Furthermore, we generate HPP with 8 vertices and apply a new method to it in order to discuss practical problems on more complex DNA-based computing. Such a larger size problem reveals more practical issues than Adleman's problems.

We present a new altanative method in section 2. Section 3 shows three problems treated in this paper. We give some discussions about practical problems on DNA-based computing based on our experiments. Finally section 5 includes concluding remarks.

2 Algorithms for HPPs for DNA-based computing

In this section, we show Adleman's algorithm, and introduce our method how we extended from his algorithm.

2.1 Adleman's experiments

Adleman's original idea how to solve the HPP is as follow:

Step 1 : Generate random paths through the graph.
Step 2 : Keep only those paths that begin with V_{in} and end with V_{out}.
Step 3 : If the graph has n vertices, then keep only those paths that enter exactly n vertices.
Step 4 : Keep only those paths that enter all of the vertices of the graph at least once.
Step 5 : If any paths remain, say "Yes"; otherwise,

say "No."

Biological operation to implement each step is described briefly:
Step 1 Each vertex i in the graph was represented O_i as a certain 20-mer sequence of DNA. Each edge $i \rightarrow j$ was represented $O_{i \rightarrow j}$ as generated that was 3' 10-mer of O_i followed by the 5' 10-mer of O_j.
Step 2 The solution of Step 1 was amplified by the Polymerase chain reaction with primers O_0 and $\overline{O_6}$.
Step 3 the product of Step 2 was run on an agarose gel (in our experiment, polyacrylamide gel). Then the 140-base pair bands which agree with the paths entering exactly 7 vertices were excised. Both Step 2 and Step 3 are repeated to intensity its purity.
Step 4 the product of Step 3 was affinity-purified with a biotin-avidin magnetic beads system to extract the paths which constructed with every vertex.
Step 5 the product of Step 4 was amplified by PCR and run on a gel.

2.2 Alternative method to analyze based on Adleman's paradigm.

As described in previous subsection, Adleman solved a HPP with 7 vertices. However it is difficult to know the details of the process how the Hamiltonian path were formed and how many populations of the Hamiltonian paths and other paths were formed.

In order to clarify these lines of detailed information, we applied new alternative method (new step 4 and 5) to solve the HPP as follows.

New Step 4: the product of Step 3 was inserted into the plasmid vector. the recombinant plasmid was introduced into *E. coli*. Since each colony contains only one species of plasmid encoding DNA sequences which represent the only one path, we can isolate and amplify each path as a form of each colony.

New Step 5: Each path was isolated from the each colony, and the DNA sequence was determined to identify each path.

While the Adleman's original method could not present how many kinds of paths were constructed in the series of reaction, this method can demonstrate all paths without excluding the non-Hamiltonian paths.

We used these new steps instead of original steps (step 4, step 5) in all experiments presented in this paper, therefore, these new steps will be simply referred as step 4, step 5 after this section.

3 Applied Problems

In our experiment, we treated these problems.

- 7 vertices HPP solved by Adleman
  In order to analyze and confirm to solve the HPP by our proposed algorithm, we repeated it which has successfully solved by him.

- simple HPP with 8 vertices:
  We consider a simple HPP with 8 vertices shown by Figure 1. This problem has only paths to form Hamiltonian path. The reason to treat this problem is as follows:

  (1) We could get to know whether our method on DNA-based computing is able to treat HPP with 8 vertices or not. If our proposed method couldn't solve that simple HPP, we cannot solve more complex HPP with 8 vertices and more.

  (2) It informs us that some samples are Hamiltonian Path in various samples. The path which was generated in this problem is the same length and constructed with the same nucleotides as next problem. Then, when we try to solve more complex HPP with 8 vertices, it works as a "positive control", for example, if it was run with other samples at the electrophoresis, same length samples are more reliable as Hamiltonian Path than simply compared at length.

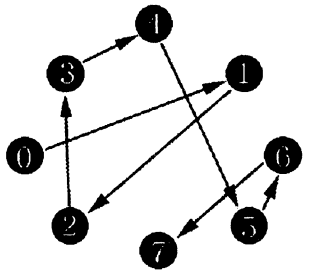


Figure 1: The simple graph which has 7 edges and one Hamiltonian path.

- HPP with 8 vertices
  It has not been discussed what problems are occurred by different size of HPP. In order to confirm that, we generated HPP with 8 vertices2. This graph has similar characteristics with 7 vertices in order not to be easier than HPP with 7 vertices.

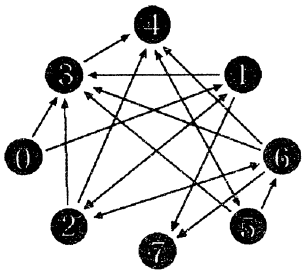


Figure 2: The generated HPP with 8 vertices.

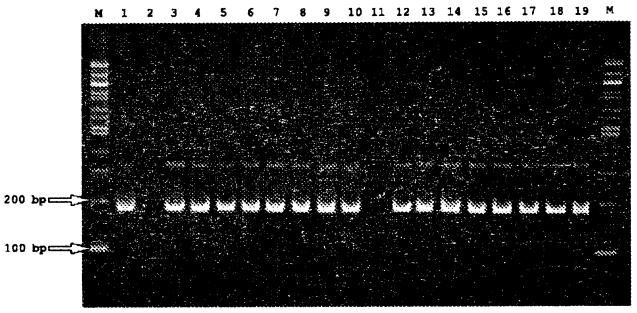


Figure 3: simple HPP with 8 vertices

4 Results

1. HPP with 7 vertices
   Table4 shows the results for HPP with 7 vertices. The shown paths were detected by determining the sequences of each DNA which were contained in randomly chosen colonies.

2. simple HPP with 8vertices
   Figure1 shows lane M, DNA size marker (100base-pair ladder). Lanes 1-19, DNA inserts amplified from 19 white or light blue transformants at **New Step 5.**

3. generated HPP with 8 vertices
   Table2 shows the results for generated HPP with 8 vertices. The result shows the length of picked colonies by electrophoresis4.

Hamiltonian paths	# of appearance
$0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$	11
Paths which doesn't exist in the graph	**# of appearance**
$0 \rightarrow 1 \rightarrow 3 \rightarrow 6$	1
$0 \rightarrow 1 \rightarrow 3 \rightarrow 0 \rightarrow 6$	1
Paths which are not Hamiltonian path	**# of appearance**
$0 \rightarrow 6$	9
$0 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$	16
$0 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$	7
$0 \rightarrow 3 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$	2
$0 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6$	3
Total	50

Table 1: DNA paths which were observed in 7 vertices problem.

Number of *E.coli* colonies picked	215
No DNA insert	37
80 bp	44
100 bp	41
120 bp	91
140 bp	2
Total	178

Table 2: Lengths and population of various inserted DNA paths in 8 vertices problem

5 Discussions

- HPP with 7 vertices
  From results, 11 DNAs include the Hamiltonian paths in the same length as Hamiltonian path length . Although we obtained the Hamiltonian paths, 32 paths were shorter and 3 paths were longer than the Hamiltonian path length. The appearance of these DNA paths was occurred by the ability of gel electrophoresis which separated DNA paths depending on their molecular weights. Furthermore, 2 DNA paths which don't theoretically exist in the graph were observed as forms of $3 \rightarrow 6$, $3 \rightarrow 0$. It means these two DNA paths are recognized as errors of DNA-based computing.

- simple HPP with 8 vertices
  As shown Figure 3, almost all samples are the same length for this problem. In fact, when DNA sequences were determined, we found these DNAs were Hamiltonian Path. It means our method could treat HPP with 8 vertices and the path which includes 8 vertices and 7 edges were generated at Step 1. This simple HPP with 8 vertices are used along generated 8 vertices HPP.

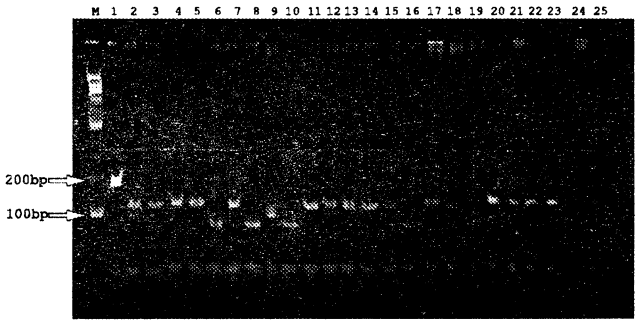


Figure 4: gel electrophoresis with generated HPP samples

- generated HPP with 8 vertices
  From results, none of DNA paths which were not only Hamiltonian path also same length as Hamiltonian path size was observed even through 178 colonies were screened(Table 2). We consider the probability to obtain Hamiltonian path as less higher for HPP with 8 vertices than for HPP with 7 vertices.

  (1) It is considerable that not enough colonies were picked and checked. Shortage of certain vertices may critical reason too, i.e. if there is a vertex which connected with many vertices, that vertex was exhausted and doesn't work as a part of Hamiltonian path.

  (2) One of other reasons is the contamination of shorter DNA paths at each steps. Although we consider that single DNA band was purified from the gels by electrophoresis, in fact, huge amount of various length of DNA paths were mingled. This phenomena appears after the purified samples were inserted into the plasmid vectors. In fact, some kinds of DNA paths were inserted into the vector plasmid (Figure 4, Table 2). We observed this phenomenon in both the 7 and 8 vertices HPP. It was observed more often in the 8 vertices HPP than 7 vertices because more various feasible solutions could be generated by the complexity in the 8 vertices problem.

  (3) PCR may promote this phenomenon. It was known that PCR amplified more shorter oligonucleotides than longer ones in the solution which includes different length of oligonucleotides.

  (4) When cloned DNA paths were amplified by colony direct PCR, one major DNA band, which was the shortest band, and several minor DNA bands were amplified and observed in gel electrophoresis (Figure 3, 4), though each *E.coli* colony should have only one species of plasmid, these DNA bands were derived from one species of DNA path used as template. This suggests that the several minor DNA bands were different from each other in their secondary structures for example, forming cruciforms or hairpins, but derived from the same DNA sequence. This phenomena might be occurred by the design of oligonucleotides we have encoded the problems.

6 Conclusions

We proposed a new alternative method for analyzing the HPP and applied it to three types of HPPs with 7 and 8 vertices HPP. We could show that which paths were generated in search paradigm. Throughout the results, we discussed some problems on DNA-based computing using *generate and search* paradigm. Some following problems were revealed by our analysis.

- Theoretically inexistent paths in the graph are were observed. This facts indicate that we have to improve the precision of DNA-based computing operations.
- Contamination of shorter DNA paths are critical problem to purify the correct size DNA paths from the gel.
- Other biological operations, such as PCR, might amplify the contamination of shorter DNA paths.

These problems will be more practical in near future, because almost all DNA-based computing adopt such biological operations[2, 3].

References

[1] L. Adleman, "Molecular Computation of Solutions to Combinatorial Problems" Science. vol. 266, pp. 1021-1024, 1994.

[2] M. Hagiya, M. Arita, "Towerds Parallel Evaluation and Learning of Boolean μ-formulas with Molecules," in Proc. 3rd DIMACS Workshop on DNA Based Computing, Pennsylvania, USA, 1997, pp. 105-114.

[3] R. J. Lipton. "DNA Solution of Hard Computational Problems," Science. vol.268, pp. 542-545, 1995.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Self-organized critical behaviors of fish schools and emergence of group intelligence

Yoshimasa Narita*, Kiyohiko Hattori**, Yoshiki Kashimori*
and Takeshi Kambara*, **
* Dept. of Applied Phys. and Chem., Univ. of Electro-Communications, Chofu, Tokyo, 182-8585 Japan
** Department of Information Network Science, Graduate School of Information Systems, University of Electro-Communications, Chofu, Tokyo 182-8585, Japan

The manuscript of this talk will be distributed at the symposium.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

The Degree of Human Visual Attention in the Visual Search

Hiroaki Mizuhara, Jing-Long Wu and Yoshikazu Nishikawa*

Department of Mechanical Engineering, Yamaguchi University
Tokiwadai 2557, Ube 755–8611, Japan
Email:wu@mechgw.mech.yamaguchi-u.ac.jp
*Faculty of Information Sciences, Osaka Institute of Technology
Kitayama 1–79–1, Hirakata 573–0171, Japan

Abstract

Human can take visual information in parallel through the vision such as the retina, but can not pay attention at the same time toward all the information. Restriction of visual attention is treated by analyzing the characteristic of visual search task. Stimuli in the visual search are thought to be processed into features at parallel in early vision. However, the authors consider that these features are not processed completely at parallel and have the reciprocal action in each other. In order to clarify the reciprocal action of the features in the visual search and the continuity of the visual attention, characteristics of reaction time are measured with changing forms of stimuli. The experimental results suggested that the reaction time is changed with changing the features. Namely, the features are affected in each other. Furthermore, the continuity of the reciprocal action are also suggested, and the degree of the visual attention is decided by the continuity. The results provided significant basic data, to propose a mathematical model of the visual attention.

Key Word: *Cognitive science, Human visual mechanism, Visual search, Visual attention, Early vision*

1. Introduction

Human can get visual information in parallel through the vision. However, human can not pay attention at the same time toward all the information. A limit of this parallel processing is the restriction of the early vision. The problem of such early vision can be treated by analyzing the characteristic of "visual search" [1].

The visual search is the task of detecting a target stimulus from various distracter stimuli. If a target is defined as distinguished from other stimuli with only one feature (feature search task), the target is found out at once, though the numbers of the stimuli are increased. If a target is defined as distinguished from distracters with conjunction of the features (conjunction search task), for example the target is defined as conjunction of color and direction, to search the target is more difficult than the feature search task and increased with the numbers of the stimuli increased. In these visual search tasks, the stimuli are thought to be processed into features in parallel in early vision [1~5]. In condition of that the stimuli have only one feature, the stimuli are thought to be processed only in early vision with no attention [3,4]. Since the stimuli are processed at parallel, the reaction time is not changed with numbers of stimuli increased. If the stimuli have plural features, the features are integrated in map of position after processed in early vision and the visual attention is directed to the map in limited suchlike spotlight [3,4]. Since frequency of the spotlight direct to map is increased with numbers of stimuli increased, the reaction time is increased.

However, some studies [1,2,5] suggested that the features are thought to be affected in each other, though the features are processed in parallel. In our previous studies [6,7], the search task of the arrow's direction showed the two types of characteristics among subjects even though in same task, the reaction time was increased or not with changing the numbers of stimuli. Furthermore, in the case of the task which thought to be processed with no attention, a study [8] reported that the visual attention is necessary even in such tasks.

Thus, we consider that visual search needs visual attention even in the condition of the target was searched at once, and the visual attention is changed continuously by the reciprocal action of the features between the condition of the target was searched at once and that the target was difficult to search. Accordingly, in order to clarify the continuity of the visual attention and the reciprocal action of the features, the reaction time and the rate of errors answered are measured.

2. Experimental Method

2.1 Experimental System

A proposed experimental system is shown in Fig. 1. Stimuli were generated by an Image Maker (IM-9800M), and presented by an Image Animator (AD-981) for observers through RGB monitor (SONY PVM1440). The stimuli were controlled by a personal computer (NEC PC-9821). The observers and the personal computer have

conversation with each other by using a response key. The experiments were carried out in a dark room.

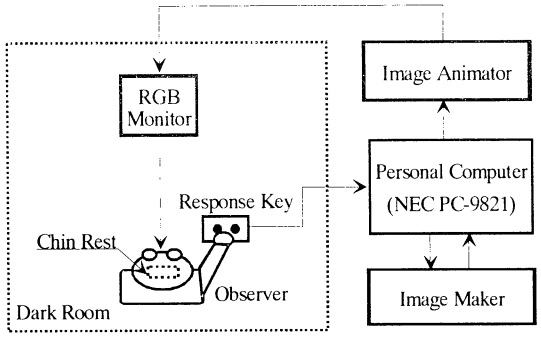


Fig. 1 Schematic diagram of the experimental system

2.2 Experimental Stimuli

Examples of the experimental stimuli are shown in Fig. 2. The tasks for observers were the search of unique direction of arrows. Numbers of the stimuli were changed with 9 (3 lines and 3 rows), 16 (4 lines and 4rows) or 25 (5 lines and 5 rows). In all experiments, the stimuli were presented centering around center of the frame (frame size : 8.5°×8.5°) with constant distance between stimuli (1.0°×1.0°).

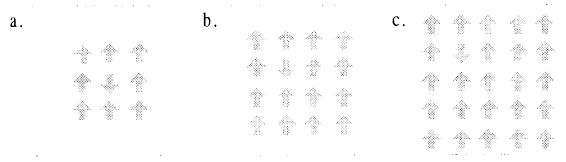


Fig. 2 Examples of the experimental stimuli with that numbers of the stimuli was (a)9, (b)16 or (c)25

The details of one stimulus(arrows) are shown in Fig. 3. In the experiment, the size of one stimulus is a constant (0.5°×0.5°) in all conditions. The arrows can be decided with the conjunction of two parts of figures, one part is a triangle and the other one is a square. Thus, to change the form of the arrows can be realized by changing the angle of triangle θ and the basic length of square l as shown in Fig. 3. In the experiment, the angle θ was varied for 53.2, 78.6, 103.9, 129.3, 154.6 and 167.3 [°], and the size of the stimuli was kept with constant(0.5°×0.5°). The basic length l was varied for 0.01, 0.1, 0.2, 0.3, 0.4 and 0.5 [°].

In the experiment, the observers were requested to indicate whether one target stimulus(arrow) was be or not, when the different types of the stimuli were presented in a random order. This arrow's direction was different from the other stimuli(arrows). During the experiment, the time between the start of presenting the stimuli and the response of the observers was measured. This time is defined as "reaction time". Time chart of the measurement is shown in Fig. 4. In the experiment, three volunteers, aged 21 ~ 25, were chosen as the observers.

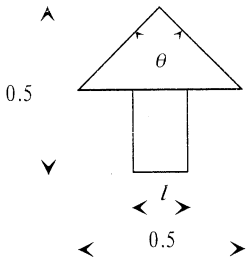


Fig. 3 Details of one experimental stimulus

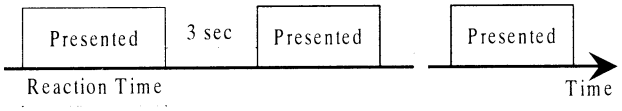


Fig. 4 Time chart of the measurement

3. Experimental Results

The experimental results in the condition of the basic length l =0.5 [°], are shown in Fig. 5. The horizontal and the vertical axes show the numbers of the stimuli and the reaction time, respectively. Each datum points is the average of 18 measurements for one observer except the errors answered.

From Fig. 5, it is can be seen that the reaction time increases as the numbers of the stimuli are increasing. In the case of θ =53.2[°](□), the average reaction time is increased from 654 to 940 [msec], when the numbers of the stimuli are varied from 9 to 25. In the case of θ =167.3[°](○), the average reaction time is increased from 1766 to 3614 [msec], when the numbers of the stimuli are varied from 9 to 25. That is to say, the reaction time is also increased with θ increased.

In order to investigate the detail of the increased speed of the reaction time, slope of the reaction time is calculated by using the least squares analysis. The unit of the slope of the reaction time is defined as millisecond per STimulus [msec/ST]. In the condition of that the target is presented (Fig. 5(a)), the slope of the reaction time in the case of θ =53.2[°](□), is 6.2 ~ 16.4 [msec/ST] for all observer. In the case of θ =167.3[°], the slope is 19.1 ~ 62.9 [msec/ST]. In the condition of the target was absented (Fig. 5(b)), the slopes are −5.6 ~ 25.1 and 60.0 ~ 130.7 [msec/ST] in the cases of θ =53.2[°](□) and θ =167.3[°](○), respectively. As explained above, these slopes are also increased with the angle θ increased.

Table 1 Rate of error answer [%] in the case of l =0.5 [°]

θ [°] / Observer	(a) Target was Presented						(b) Target was Absented					
	53.2	78.6	103.9	129.3	154.6	167.3	53.2	78.6	103.9	129.3	154.6	167.3
(1) H.H	0.00	0.00	0.00	0.00	7.41	33.3	0.00	0.00	0.00	1.85	0.00	1.85
(2) H.Y	0.00	0.00	0.00	0.00	0.00	16.7	0.00	0.00	0.00	0.00	0.00	0.00
(3) O.N	7.41	9.26	0.00	0.00	7.41	22.2	3.70	5.56	7.41	7.41	9.26	3.70

Table 2 Rate of error answer [%] in the case of θ =167.3 [°]

l [°] / Observer	(a) Target was Presented						(b) Target was Absented					
	0.01	0.1	0.2	0.3	0.4	0.5	0.01	0.1	0.2	0.3	0.4	0.5
(1) H.H	1.85	7.41	9.26	27.8	40.7	33.3	0.00	0.00	1.85	7.41	0.00	1.85
(2) H.Y	1.85	0.00	5.56	5.56	7.41	16.7	0.00	0.00	5.56	0.00	0.00	0.00
(3) O.N	5.56	1.85	14.8	20.4	22.2	22.2	1.85	7.41	14.8	5.56	11.1	3.70

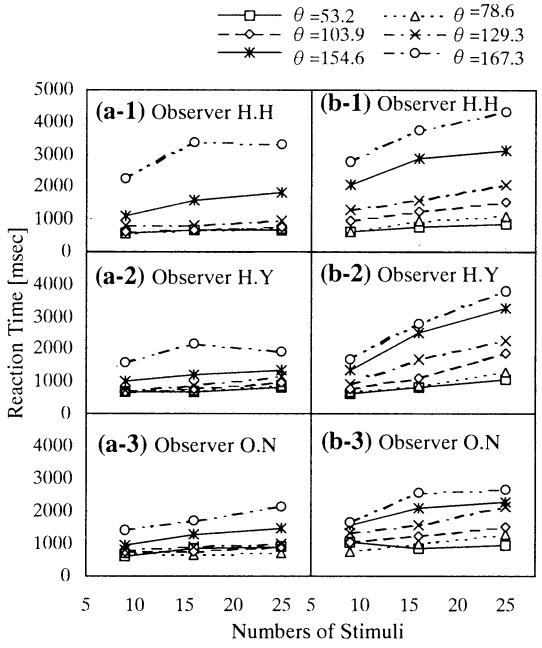


Fig. 5 Reaction time as a function of the numbers of the stimuli in the condition of that the angle θ was changed and the basic length l was 0.5 [°]. Target was (a)presented or (b)absented for observers (1)H.H, (2)H.Y and (3)O.N.

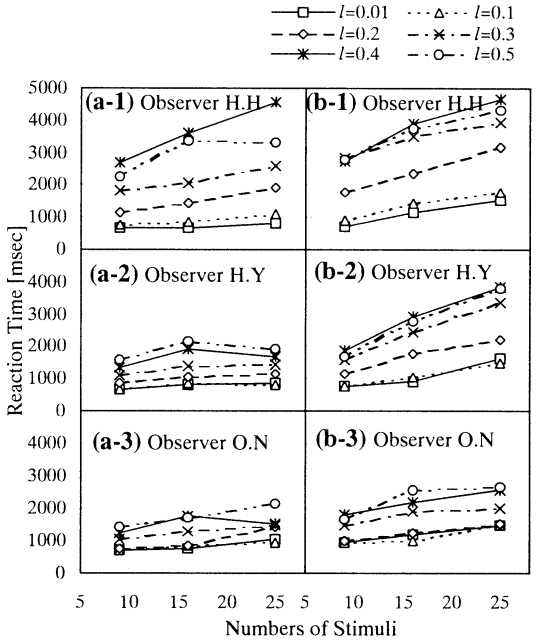


Fig. 6 Reaction time as a function of the numbers of the stimuli in the condition of that the basic length l was changed and the angle θ was 167.3 [°]. Target was (a)presented or (b)absented for observers (1)H.H, (2)H.Y and (3)O.N.

The rates of error answer in the same condition of Fig. 5, are shown in Table 1. In the condition of that the target was presented(Table 1(a)), the rate in the case of θ =167.3 [°] is larger than other cases remarkably, for all observers. Whereas, in the condition of that the target was absented(Table 1(b)), a significant change of the rate of errors answer is not shown.

Figure 6 shows the experimental results in the condition of θ =167.3[°] with changing the basic length l. In condition of the target was presented(Fig. 6(a)), the slope of the reaction time is 8.8 ~ 20.7 [msec/ST] in the case of l =0.01[°], and is 19.1 ~ 62.9 [msec/ST] in the case of l =0.5[°]. Incidentally, the condition of l =0.5[°] in Fig. 6 , is the same experiment as the condition of θ =167.3[°] in Fig. 5. In condition of the target was absented(Fig. 6(b)), the slope is 30.9 ~ 55.7 [msec/ST] in the case of l =0.01[°]. Therefore, in changing the condition of the l , these slopes are increased. This peculiarity is similar to the condition of that the θ is changing.

The absolute value of the reaction time is increased from 718 to 1217 [msec] in the case of l =0.01[°], and is increased from 1766 to 3614 [msec] in the case of l =0.5[°]. The rate of the error answer in this condition is shown in Table 2. In the condition of that the target was

presented(Table 2(a)), the average of the rate of error answer is larger than other conditions as shown in Tables 1 and 2.

4. Discussion

In order to clarify the reciprocal action of the features and the continuity of the visual attention, the results of that the slope and the absolute value of the reaction time were changed continuously with changing the forms of stimuli, is a key point. Figures 7 and 8 show the slope and the absolute value of the reaction time as a function of the angle θ. In Fig. 7, one datum point shows that the slope of the reaction time is the average value of the three observers. The slope is calculated by using the least squares analysis. In Fig. 8, one datum point shows that the absolute value of the reaction time is the average value of the three observers of in the three cases of that the numbers of the stimuli are set at 9, 16 or 25.

In the condition of that the target was presented(Fig. 7(a)), the slopes are almost constant in the cases of l =0.01[°] and l =0.1[°]. In the other cases of that the l is larger than 0.2[°], the slopes increase with the θ increased. In the condition of the target was absented(Fig.7(b)), the increasing of the slope is more rapid than in the cases showed in Fig.7(a). As shown in Fig. 8, the absolute value of the reaction time is increased, when the angle θ is increasing. This peculiarity is similar to the characteristic of the slope as shown in Fig. 7.

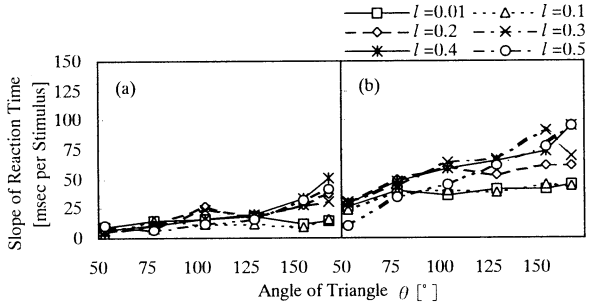


Fig. 7 Slope of the reaction time as a function of the angle θ. Target was (a) presented or (b) absented.

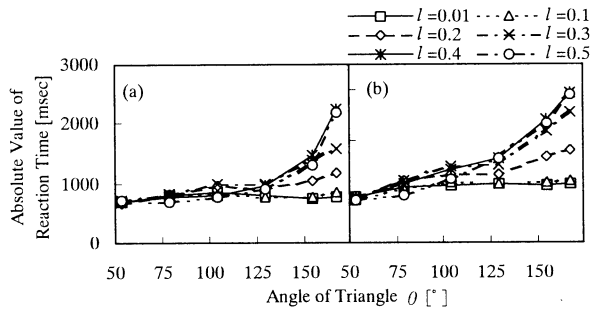


Fig. 8 Absolute value of the reaction time as a function of the angle θ.Target was (a) presented or (b) absented.

These results suggested that the slope and the absolute value of the reaction time are affected by both of the θ and the l. If the θ and the l are processed in parallel, the slope of the reaction time is constant with changing the l in the condition of that the θ is constant. However, the experimental results show that the slope of the reaction time is changed in the cases of θ=154.6 and 167.3[°], when the l is varying. Therefore, the features of the stimuli are affected in each other. As shown in Fig. 8, the absolute value of the reaction time is increased with the reciprocal action of features are changed continuously. The results suggested that the reciprocal action of features decides the degree of the visual attention continuously. When the reciprocal action of the features is extremely small, the slope of the reaction time shows the characteristic of the feature search task.

5. Conclusion

In order to clarify the reciprocal action of the features in the visual search and the continuity of the visual attention, the characteristics of the reaction time have been measured with changing the forms of the stimuli. The experimental results suggested the characteristics of the reaction time are changed continuously with the forms of the stimuli changing. Namely, the features are affected in each other. Furthermore, the continuity of the reciprocal action decides the degree of the visual attention.

Based on the experimental results a mathematical model of the degree of the visual attention is needed to be proposed, in our future work.

References

1. Yokosawa K (1994), Multiresolutional Model for Analysis of Visual Attention and Visual Search Performance (in Japanese). *Japan Cognitive Science* **1**(2) : 64–80.
2. Yokosawa K, Kumada T (1996), Visual Search —Phenomena and Processes (in Japanese), *Japan Cognitive Science* **3**(4) : 119–138.
3. Treisman A (1986), Feature and Objects in Visual Processing, *Scientific America* **254**(11) : 114–124.
4. Treisman A (1992), Spreading Suppression or Feature Integration? A Reply to Duncan and Humphreys(1992), *Journal of Experimental Psychology:Human Perception and Performance* **18**(2) : 589–593.
5. Humphreys G.W, Müller H.J (1993), SEarch via Recursive Rejection (SERR):A Connectionist Model of Visual Search, *Cognitive Psychology* **25**, 43-110.
6. Wu J.L, Mizuhara H, Nishikawa Y (1998), Measurement of Human Visual and Auditory Characteristics for A Virtual Driving System, *Proc. of AROB* Ⅲ : 715–718.
7. Wu J.L, Mizuhara H, Nishikawa Y (1999), On Human Related Characteristic between Visual Search and Speach Perception (in Japanese), *Transactions of the Institute of Systems, Control and Information Engineers* **12**(2) : (be in press).
8. Joseph J.S, Chun M.M, Nakayama K (1997), Attentional Requirments in a 'Preattentive' Feature Search Task, *Nature* **387** : 805–807.

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

Human Interactive Characteristic between Binocular Disparity and Occlusion for Depth Perception

Jing-Long Wu and Hiroyasu Yoshida

Department of Mechanical Engineering, Yamaguchi University
Tokiwadai 2557, Ube 755-8611, Japan
Email:wu@mechgw.mech.yamaguchi-u.ac.jp

Abstract

Recently, the Head Mounted Display(HMD) has been widely used in virtual reality technology. In order to develop a HMD with high reality-performance, how to present the depth information is very important. In the HMD, the depth information is mainly obtained from the cues of the binocular disparity and the occlusion. Therefore, human interactive characteristic between the binocular disparity and the occlusion for depth perception is needed to investigate quantitively. In this study, the depth perceptive threshold of binocular disparity and occlusion are measured. The experimental results suggested that if the binocular disparity is set at a proper value, the depth information is obtained from the cue of the binocular disparity mainly, and if the occlusion ratio larger than a threshold value the depth information is obtained from the cue of the occlusion. Based on the experimental results, we can find a method to make images with depth information in the HMD, when the cues of the binocular disparity and the occlusion are used.

1. Introduction

The Head Mounted Display (HMD) as a presented device of the visual information, has been widely applied in the virtual reality technology[1]. In the HMD, the depth information is mainly obtained from the cues of the binocular disparity and the occlusion. We can see an image with depth information through HMD when a proper binocular disparity and/or the occlusion are set. In the binocular vision, the disparity is a main cue to obtain depth perception. In the monocular vision, there are many cues (i.e., occlusion, shadow, different size and so on) to obtain depth perception[2]. In the previous studies[3][4], the relationship of cues of depth perception has been investigated. In this study, we pay attention to the cues of the binocular disparity and the occlusion for the depth perception. In the HMD, the concept of that the depth perception is obtained from the cues of the binocular disparity and the occlusion can be explained by Fig. 1. As shown in Fig. 1(a), we can see that the black circle pattern appears in front of the white circle pattern when the binocular disparity between the images of the left eye and the right eye is set at zero. In this case, the depth perception can be obtained from the cue of the occlusion only. However, we can perceive a stereo view if a proper binocular disparity between the images of the left and the right eyes is set as shown in Fig.1(b). That is to say, we can see the white circle appears in front of the black circle even though the cue of the occlusion has an inverse effect (i.e., the black circle pattern appears in front of the white circle). In this case, the depth perception can be obtained from the cue of the binocular disparity.

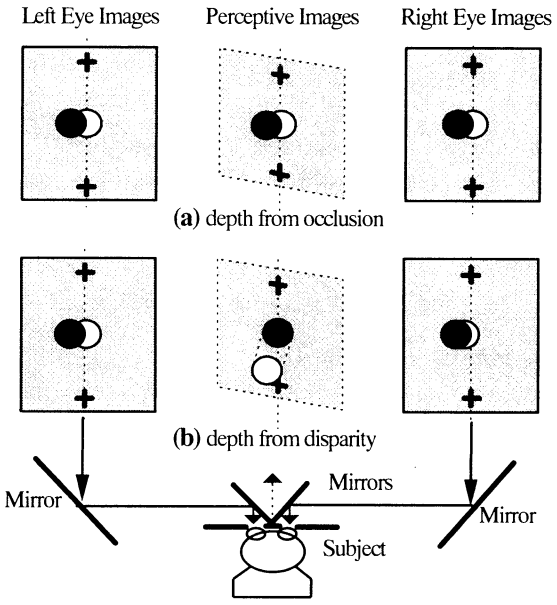


Fig.1 The depth information from binocular disparity and occlusion in the HMD

They are very important problems to realize stereo view with high reality performance, and to investigate the mechanism of the human depth perception. In order to solve those problems, the interactive characteristic between the binocular disparity and the occlusion for depth perception is measured in this study. The experimental results suggested that if the binocular disparity is set at a proper value, the depth information is obtained from the cue of the binocular disparity mainly, and if the occlusion ratio is larger than about 0.91 the depth information is obtained from the cue of the

occlusion. Based on the experimental results, we can find a method to make images with depth information in the HMD when the cues of the binocular disparity and the occlusion are concurrently used.

2. Experiment Method

2.1 Experimental System

A developed experimental system is shown in Fig.2. The stimuli were made by an image maker (IM-9800M), and were shown by the image maker and an image animator (AD-981) for the left and the right eyes of the subjects through two RGB monitors (SONY-PVM1440). The stimuli of the left and the right eye could be controlled by a personal computer (NEC PC98-21). The subjects could fuse the stimuli by using two mirrors. And, the answer of the subject was taken by the personal computer, when a response key is pressed. The experiment was carried out in a dark room.

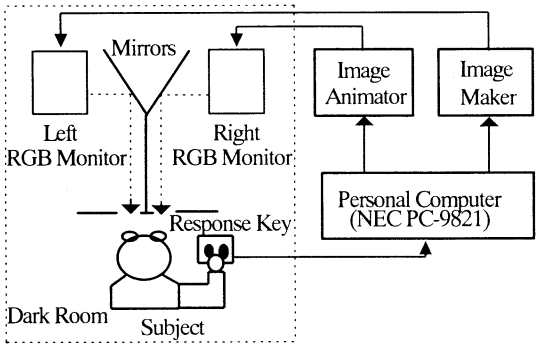


Fig. 2 Outline of experimental system

2.2 Experimental Stimuli

The experimental stimuli of the left and the right eye are shown in Fig.1, they were concurrently presented to the left and the right eye independently. The target stimulus was composed by one white circle and one black circle patterns per eye. And the white circle pattern was occluded by the black circle pattern. In the target stimulus, the positions of both the white and the black circles are shown in Fig. 3. A ratio (R) of the occlusion is defined as $R = S_1/S_2$. Where, S_1 is the part area (shown by the horizontal stripes) of the white circle, it is occluded by the black circle as shown in Fig. 3. S_2 is the area of the whole white circle. In the experiment, the measurement was curried out for three cases. They are the cases of that the primal occlusion ratio (R_0) is set at 0.39, 0.49 or 0.58.

The size of background stimulus was 5.0°×7.0°. The size of the crossed stimulus was 0.5°×0.5° and it can assisted subjects to fuse the background stimuli of the left and the right eye. The size of the circle pattern (both the white and the black) of the target stimulus was 1.2° in diameter. The brightness of the white circle was 25.2 cd/m^2, and that of the black circle was 0.13 cd/m^2. Previous study suggested that red and blue colors have the depth cue of hue[3]. In order to avoid the depth cue of hue, the target stimulus used white and black, and the background stimulus used gray .

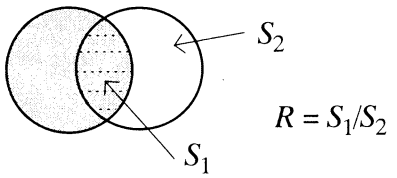


Fig. 3 Definition of the occlusion ratio in the target stimulus

2.3 Experimental Procedure

(1) The subjects sat in the dark room, and the experiments started after the initial 6 minutes of the dark adaptation.

(2) The subjects made effort to fuse the background stimuli of the left and the right eye with the help of fixation points (crossed stimulus).

(3) The subjects looked in the background stimuli for light adaptation.

(4) In the measurement, the position of the black circle was fixed, and the position of the white circle of the right eye could be moved to the black circle in order to vary the binocular disparity. When the stimuli with different binocular disparity were presented in a random order, the subjects were required to answer that the white circle was in front of the black circle or the black circle was in front of the white circle. The time chart of the measurement is shown in Fig. 4[5].

2.4 Subjects

The subjects were three senior students and their conditions are shown in Table 1.

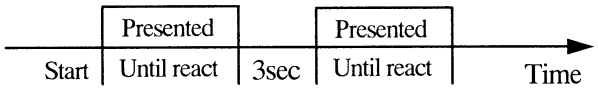


Fig. 4 The time chart of measurement

Table 1 Subjects character

Subject	Age	Sex	Visual Acuity	Dominant Eye
H.H	25	male	R:0.4, L:0.6	Right
O.N	21	male	R:1.2, L:1.2	Left
H.Y	22	male	R:1.2, L:1.5	Right

3. Experimental Results

The experimental results of the three subjects are shown in Fig. 5. The horizontal axes show the binocular disparity, the vertical axes show the depth perceptive percentage of the white circle. This is the

percentage of subject's answering that the white circle is in front of the black circle. Each datum is the average of thirty measurements for one subject.

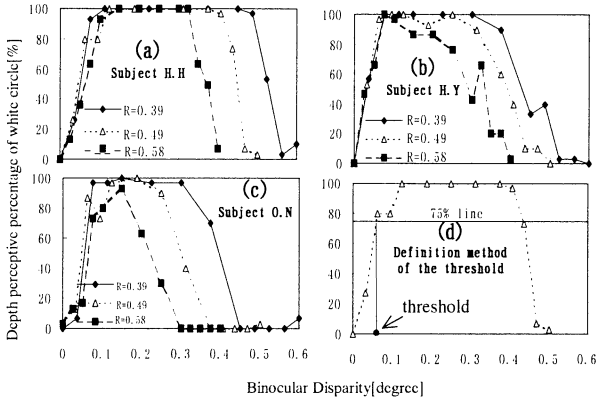


Fig. 5 The depth perceptive percentage of the white circle as a function of the binocular disparity

From Fig. 5, it can be seen that the depth perceptive percentage of the white circle increases as the binocular disparity is increasing when the binocular disparity has small value, and the percentage begins to be 100(%) when the disparity is 0.07~0.19(degree). This results mean that the depth perception is enhanced by the binocular disparity. When the disparity is more increased, the percentage decreases in the range of the large disparity. In this case, the cue of the occlusion is dominant than the cue of the binocular disparity, to contribute depth perception. Because the occlusion ratio increases when the binocular disparity is increasing.

In psychological studies, the threshold was generally defined at the percentage values with 75%. In this paper, we define the depth perceptive threshold of the white circle with the binocular disparity at the percentage values with 75% . As an example, the definition method is explained in Fig. 5(d). In this way, the thresholds can be calculated from Figs. 5(a), (b) and (c), and they are summarized in Table 2. As shown in Table 2, the threshold of the depth perception is between 0.045 and 0.16 degree, and its average is about 0.084 degree. That is to say, if the binocular disparity is large than about 0.084 degree we can see a stereo view as shown in Fig. 1(b).

Table 2 Depth perceptive threshold of binocular disparity (degree)

Subject	Experimental kinds			Average
	R_0=0.39	R_0=0.49	R_0=0.58	
H.H	0.12	0.058	0.16	0.11
O.N	0.063	0.053	0.15	0.089
H.Y	0.052	0.045	0.059	0.052
Average	0.078	0.052	0.12	0.084

4. Discussion

In order to investigate the detail of the cue of occlusion for the depth perception, the depth perceptive percentages of the black circle pattern can be reploted from Fig. 5. The reploted results are shown in Fig. 6. The horizontal axes show the occlusion ratio, the vertical axes show the depth perceptive percentages of the black circle pattern. This is the percentage of subject's answering that the black circle is in front of the white circle.

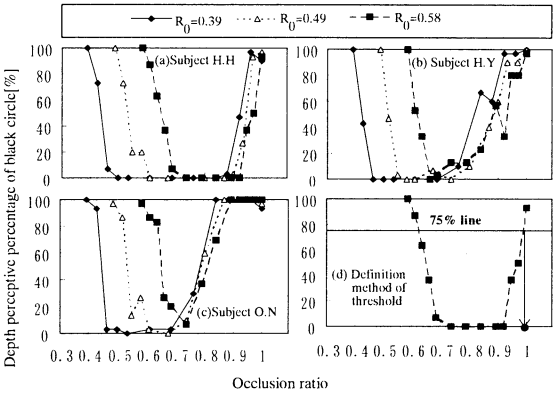


Fig. 6 The depth perceptive percentage of the black circle as a function of the occlusion ratio

The main points of Fig. 6 can be summarized in three cases which are the ranges of the small, the middle and the large occlusion ratio. (1) In the range of small occlusion ratio, the depth perceptive percentage of the black circle has large value. In this case, the binocular disparity is set at small value, so the cue of the occlusion is advanced to perceive depth information of the black circle; (2) In the range of middle occlusion ratio, the depth percentage of the *black circle* has small value. In this case, the binocular disparity is set at a lager value, so the cue of the binocular disparity is advanced to perceive the depth information of the *white circle*. (3) In the range of large occlusion ratio, the depth percentage of the black circle has large value. In this case, the binocular disparity is also set at a large value, the results mean that if the occlusion ratio is larger than a threshold value the cue of the occlusion is strong to advance to perceive depth information even though the binocular disparity is set at a large value. Same as it is explained in Chapter **3**, we define the depth perceptive threshold of the black circle with the occlusion at the percentage values with 75% . As an example, the definition method is explained in Fig. 6(d). In this way, the thresholds can be calculated from Figs. 6(a), (b) and (c), and they are summarized in Table 3. As shown in Table 3, the threshold of the depth perception is between 0.82 and 0.99, and its own average is about 0.91. That is to say, if the

occlusion is larger than about 0.91 we can see depth information as shown in Fig. 1(a), the black circle is in front of the white circle.

Table 3 The depth perceptive threshold of occlusion ratio

Subject	Experimental kinds			Average
	R_0=0.39	R_0=0.49	R_0=0.58	
H.H	0.96	0.97	0.99	0.97
O.N	0.82	0.83	0.85	0.83
H.Y	0.90	0.93	0.96	0.93
Average	0.89	0.91	0.93	0.91

In order to investigate that the primal occlusion ratio how to effect the depth perceptive threshold, the experimental results of Tables 2 and 3 are plotted in Figs. 7(a) and (b), respectively. The horizontal axes show the primal occlusion ratio. The vertical axes are the depth perceptive thresholds of the binocular disparity and the occlusion in Figs. 7(a) and (b), respectively. From Fig. 7, it is can be seen that the depth perceptive thresholds are not remarkably changed when the primal occlusion ratio(R_0) is varying from 0.39 to 0.58. That is to say, the depth perceptive threshold is not dependent on the primal condition of the occlusion ratio.

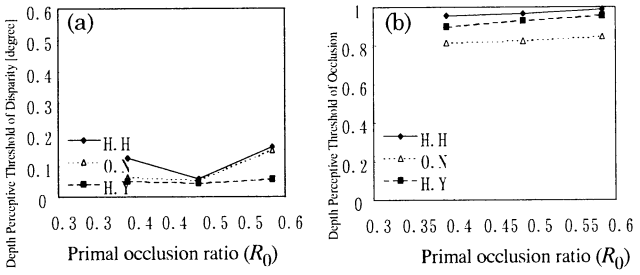


Fig. 7 Relationship of the depth perceptive threshold and the primal occlusion ratio

As discussed above, our experimental results suggested that the depth information is mainly from the cue of occlusion if the occlusion ratio is larger than about 0.9 or the binocular disparity is smaller than about 0.08 degree. Based on this psychological experimental results, we can propose a presenting method of the depth information in the HMD. To present depth information, we can use cues of the binocular disparity and the occlusion. If the occlusion ratio is large than 0.9 or that the binocular disparity is smaller than about 0.08 degree, the binocular disparity is not needed to set. In the others, to set a proper binocular disparity is a useful way to present the depth information in the HMD.

5. Conclusion

In order to realize a visual view with high depth information in the HMD, the interactive characteristic between the binocular disparity and the occlusion for depth perception, has been measured. The experimental results suggested that if the binocular disparity is set at a proper value, the depth information is obtained from the cue of the binocular disparity mainly, and if the occlusion ratio is larger than about 0.9 the depth information is obtained from the cue of the occlusion. Based on the experimental results, we could find a method to make images with depth information in the HMD, when the cues of the binocular disparity and the occlusion are concurrently used.

In our future work, the sensitivity of the depth perception is needed to be measured, in order to investigate the mechanism of the human depth perception.

Acknowledgments

We would like to express our gratitude to Mr. Hiroaki Mizuhara for his help in the programming of the experiment. The authors are also most grateful to the subjects for their cooperation in the experiment.

References

[1] Jing-Long Wu, Masaomi Nakahata, Sadao Kawamura, Ikuko Nishikawa and Hidekatsu Tokumaru, "Measurement of Binocular Stereoacuity for Design of Head Mounted Display with Wide View", Proc. of IEEE International Conference on Systems, Man and Cybernetics, Vol.2, pp.929-934(1996)

[2] Tadasu Ooyama, Shogo Imai and Tenji Wake, "A new version of sensational and perceptive psychological handbook"(in Japanese), Seishin Shobou, (1994)

[3] Satoshi Shioiri and Takao Sato, "Interaction between shading and stereopsis for depth perception"(in Japanese), ITEC'90, pp.177-178(1990)

[4] Satoshi Shioiri and Takao Sato, "Binocular stereo view and Monocular transaction" (in Japanese), Television, Vol.45, No.4, pp.431-437(1991)

[5] Jing-Long Wu, Masaomi Nakahata and Sadao Kawamura, " A New Head Mounted Display System with Adjustable Disparity for High Depth-Performance", Proc. of IEEE International Conference on Systems, Man and Cybernetics, Vol.1, pp.298-303(1995)

Proc. of The Fourth Int. Symp. on Artificial Life and Robotics (AROB 4th'99),
Oita, Japan, 19-22, January, 1999,

A Following-Type Force Display for the Virtual Catch Ball System

Koichi Kimura, Jing-Long Wu, Masayuki Kitazawa and *Yoshiro Sakai

Department of Mechanical Engineering, Yamaguchi university
*Department of Kansei Design and Engineering, Yamaguchi university
Tokiwadai 2755, Ube, Yamaguchi 755－8611, Japan
E-mail:wu@mechgw.mech.yamaguchi-u.ac.jp

Abstract

In our previous study, a virtual catch ball system was designed. The virtual catch ball system consists of pitch ball, catch ball and Head Mounted Display (HMD) subsystems. For the catch ball subsystem, the force display of simple structure, high safety and low cost is required. Some kinds of force display systems were already proposed by other researches so far. However, the motion range of those force displays is considerably small, or the large mechanical parts are necessary to attain enough motion space.

In this study, the authors proposed a following-type force display for the virtual catch ball system. The force display consists of a force unit, a rotary motion unit and a vertical motion unit. The proposed force display can generate the force in the different directions of the vertical and the horizontal plane, and the merits of this system are large motion range, high safety and low cost. The performance of the proposed force display is demonstrated by several basic experiments.

Key words: *Virtual reality, Virtual catch ball system, Force display*

1. Introduction

Virtual reality technology has been widely applied in telerobotics, sports training and so on. For virtual systems, a variety of visual displays and auditory displays have been developed. In many cases of the virtual reality and the telerobotics, the force information is very important. In general, it is difficult to present force information. Because in these systems, simple structure, high safety and low cost are needed. Up to now, several force display systems have been already proposed so far[1-3]. However, the motion ranges of those force displays, are rather small. As most of the force display systems have multilink, large mechanical parts are required in order to guarantee a large motion space [4, 5].

As an application of the virtual reality, we have been designed a virtual catch ball system[6]. The concept of the virtual catch ball system is illustrated in Fig. 1. The system consists of three sub systems; i.e., a pitch ball subsystem, a catch ball subsystem and a HMD subsystem with 3D images. HMD subsystem has been developed by our previous studies[7,8].

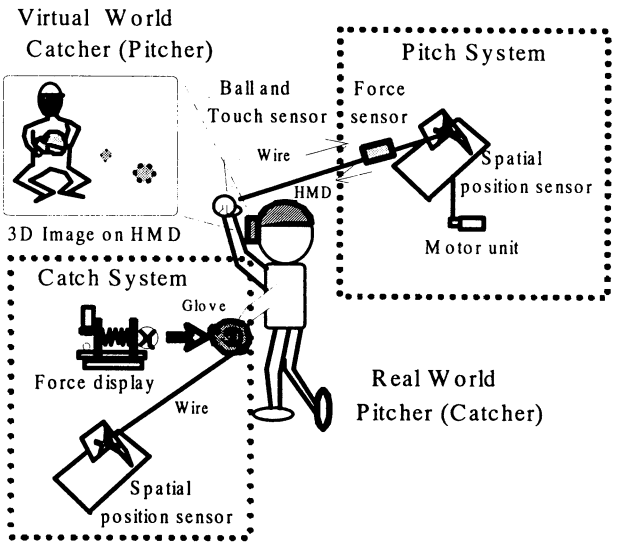


Fig. 1 Concept of the virtual catch ball system

In the pitch ball subsystem, the position of a ball is determined by a wire type spatial position sensor in real-time. The wire type spatial position sensor was developed by our previous study[9]. It has characteristics of simple structure, large measurement area, noiseless and low cost. After the real world pitcher's pitching the ball, it is rewound by a motor unit through a wire of the spatial position sensor for next play. The locus of the ball is calculated by its position and its speed at the time when the ball is released from the hand of the real world pitcher, and then the 3D images of the ball locus and the virtual catcher are presented for real world pitcher through HMD as visual information. The tension of the wire is controlled by force feedback loops using a personal computer through an Analogue to Digital (A/D) board and a force sensor.

In the catch ball subsystem, when the real world catcher catches a ball, the force is presented by a force display. At the same time, the 3D images of the virtual pitcher, the glove of the real world catcher and the locus of the ball are

presented by the HMD subsystem. The position of the glove is measured by a spatial position sensor in real-time. The structure of this spatial position sensor is similar to another one in the pitch subsystem. When the virtual pitcher pitches a ball on HMD, the real world catcher moves his hand(glove) to catch it. If the position of the glove is in agreement with the locus of the virtual ball images, the force display pitches a ball. Then the real world catcher can catch the ball successfully.

The 3D images of the virtual catcher (pitcher), the locus of the ball and the glove of the real world pitcher (catcher) are presented by the HMD subsystem. In the virtual catch ball system, a simple force display has been developed[8]. The simple force display has the problem of that the force direction can not be changed to follow the user's hand movement. In this study, based on the results of our previous study, the authors develop a following-type force display.

The following-type force display can follow the motion of the user's hand to present the force in different direction. The proposed force display has characteristics of simple structure, large movable area, noiseless and low cost. The performance of the following-type force display is demonstrated through several basic experiments.

2. A Following-Type Force Display

2.1 Concept of the following-type force display

The concept of the following-type force display can be explained in Fig.2. In the force display, the force is realized

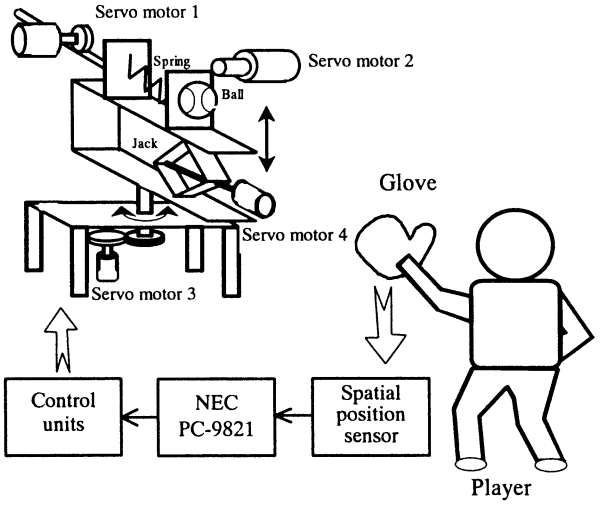


Fig. 2 Concept of the following-type force display

by using a spring. The spring is compressed by servo motor 1. When the spring returns (the returning time is controlled by servo motor 2) to its original position, the ball is pitched. The horizontal and the vertical direction of the pitched ball can be adjusted by servo motor 3 and servo motor 4, respectively. The value of the force and the directions of the pitched ball are controlled by a personal computer. When the player moves his/her hand, the position of the glove is measured by the spatial position sensor developed by our previous study[9]. The flying direction of the pitched ball is determined to agree with the glove position. That is to say, the force display can follow the player glove. This force display is called the following-type force display.

2.2 Structure of the following-type force display

The structure of the proposed following-type force display is shown in Fig. 3. The force display consists of a force unit, a vertical motion unit and a rotary motion unit.

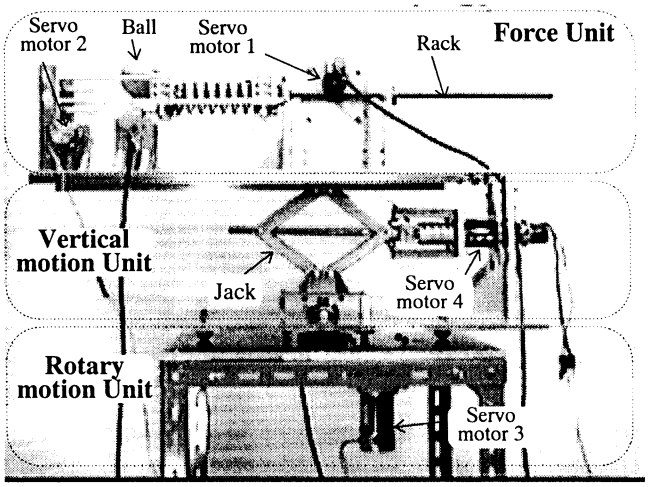


Fig. 3 Overall view of the following-type force display

(a) Structure of the force unit

As shown in Fig. 4, the force is generated by a compressed spring(0.2270 kgf/mm). Rotary motion of the servo motor 1(SANYO M605-001) is converted to a compressing motion of spring with the gear 1 and the rack. The compression stroke of the spring is measured by the potentiometer 1, and is controlled by a personal computer through the servo motor 1 in order to generate different levels of the force. The compressed spring is stopped by a pick. If the compression of the spring reaches a set level, the servo motor 1 stops and the servo motor 2 (INC RH-11-3001-SP) starts. Then the wire is rolled back with a wire drive pulley, and the pick is come off by rolling wire. At this time, the force is generated by the spring's returning to its original position.

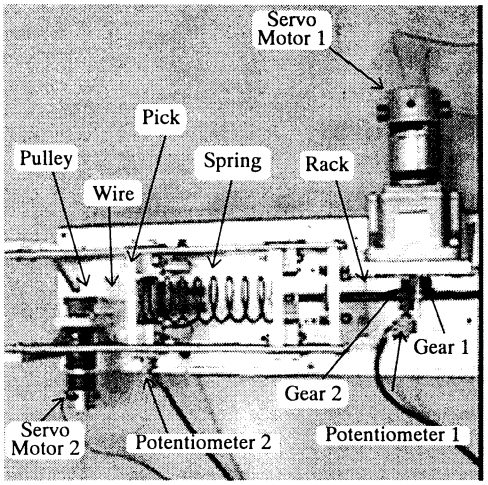


Fig. 4 Structure of the force unit

(b) Structure of the vertical and the rotary motion unit

As shown in Fig. 3, the force unit was set on the vertical motion unit. The force unit and the vertical motion unit are connected to each other by a joint, and they can be rotated by a rotary motion unit. The structure of the vertical and the rotary motion unit is shown in Fig. 5. Rotary motion of the servo motor 4 (INC RH-14-6002-E100AO-SP) is converted to the vertical motion by using a jack(a car jack), sliding rail and joint mechanism.

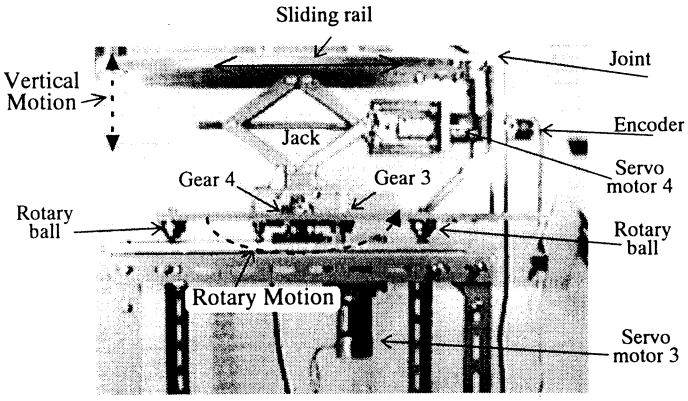
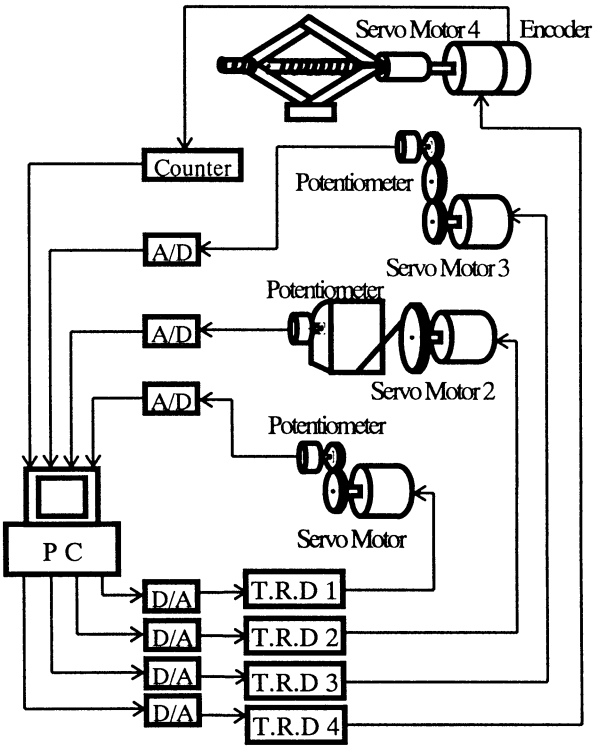


Fig. 5 Structure of the vertical and the rotary motion unit

The rotary motion is realized by servo motor 3(SANYO M605-001) through the mechanism of a gear set (Gear 3 and Gear 4) and rotary balls. When the servo motor 4 is driven, the force unit and the vertical motion unit can be rotated. The gear set is used to change the rotary speed, and the rotary balls are used to support the rotary motion smoothly. The vertical and the rotary motion of the following-type force display can be concurrently controlled by the personal computer.

2.3 Whole system of the force display

The block diagram of the whole system of the following-type force display is shown in Fig. 6. In the system, the force unit consists of two servo motors(1 and 2), two potentiometers (1 and 2), control units(T.R.D 1 and 2) and A/D(D/A) boards. The units of the rotary and the vertical motion are constructed by mechanisms of the servo motors 3 and 4, respectively. The force level and the force direction of the force display are controlled by the personal computer (NEC PC-9821).



T.R.D : TITECH ROBOT DRIVER

Fig. 6 Block diagram of the proposed following-type force display

3. Performance of the Proposed Force Display

The performance of the proposed following-type force display is demonstrated by several basic experiments. The motion ranges of the force display are 27 and 360 degrees in the vertical and the horizontal direction, respectively. The static and the dynamic characteristics of the rotary motion unit are demonstrated. The performance of the vertical motion unit, the force unit and the whole system must be evaluated in our future work.

3.1 Static characteristic

The static characteristic of the rotary motion unit is shown in Fig. 7. The horizontal axis shows the target angle

that is commanded by the personal computer. The vertical axis shows the rotary motion angle of the rotary motion unit. As shown in Fig. 7, the measurement value is extremely similar to the target value. So, it can be said that the proposed rotary motion unit has good static performance in order to present the force with different direction.

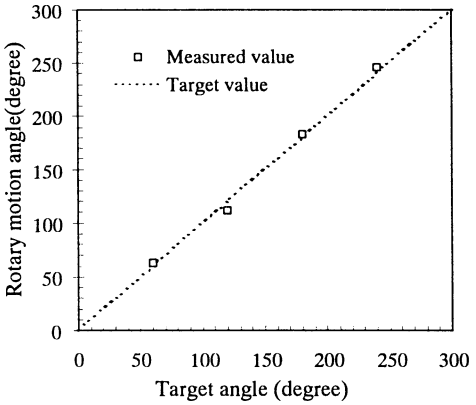


Fig. 7 Static characteristic of the rotary motion unit

3.2 Dynamic characteristic

The dynamic performance of the rotary motion unit is evaluated applying a sine signal as the input to the unit. The results are shown in Fig. 8. The horizontal axis shows the elapsed time. The vertical axes of the left and the right side show the input and output value of the rotary motion unit, respectively. From Fig. 8, it is can be seen that the rotary motion unit can follow the input signal smoothly. The delay time is about 0.4 second. It can be said that the rotary motion unit has sufficient dynamic performance for the virtual catch ball system.

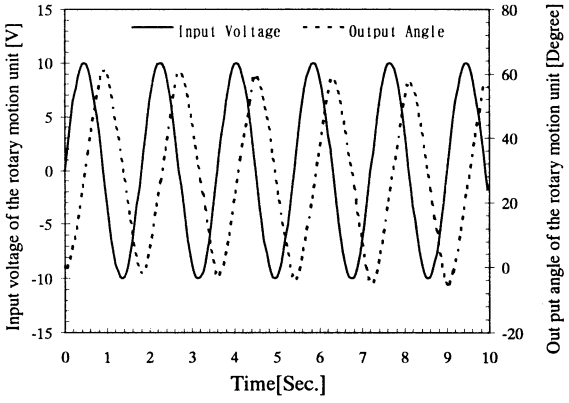


Fig. 8 Dynamic characteristic of the rotary unit

4. Conclusion

In this study, the following-type force display was proposed for the virtual catch ball system. The proposed force display is constructed by the force unit, the rotary motion unit and the vertical motion unit. The three units can be concurrently controlled by the personal computer. The performance of the rotary motion unit is demonstrated by several basic experiments, and found to be good enough. The results suggested that the proposed following-type force display has sufficient performance for the virtual catch ball system. Our future work is to implement the following-type force display to the virtual catch ball system.

References

1. Takeda T, Tsutsui T, Sato M and Kukimoto N (1994), "Computer Simulation of Basketball Dribble Using Force Display (in Japanese)", Human Interface, Vol.9,N&R : 241-246
2. Ohkura M, Yasukawa T, Kawamoto H, Hirosawa T, Hayakawa M, Tanaka K and Miyake H (1995), "Evaluation of force display for virtual reality (in Japanese)", Technical Report of Ieice. MBE95-86 (1995-09) : 75-80
3. Iwata H, Nakagawa T and Yano H (1992), "Development of a force display for Large Working Volume (in Japanese)", Human Interface 42-9 : 57-64
4. Simizu S, Ito S et al. (1993)"A Basic Study of a Force Display using a Metal Hydride Actuator". Tthe 2nd IEEE International Workshop on Robot and Human Communication. 211-215
5. Collins C. L and Long G. L.(1995) "The Singularity Analysis of an In-Parallel Hand Controller for Force-Reflected Tleoperation". IEEE Trans. on Robotics and Automation. **11**(5):661-669
6. Wu J-L, Kitazawa M, Kimura K and Sakai Y(1998) "Human Force Perceptive Characteristic and A Force Display for Virtual Catch Ball System", International Conference on Advanced Mechatronics. **1**:133-113
7. Wu Jing-Long, Nakahata M, Kawamura S et al. (1996) "Measurement of Binocular Stereoacuity for Design of Head Mounted Display with Wide View". IEEE International Conference on Systems, Man and Cybernetics. **2**: 929-934
8. Wu Jing-Long, Nakahata M and Kawamura S(1995) "A New Head Mounted Display System with Adjustable Disparity for High Depth-Performance". IEEE International Conference on Systems, Man and Cybernetics. **1**:298-303
9. Kitazawa M, Wu Jing-Long and Sakai Y(1998) "A Spatial Position Sensor for a Virtual Catch Ball System". International Conference on Advanced Mechatronics. **1**:139-144

Author Index

A

Abe, K. 484, 488, 492, 496, 500, 737, 741
Adachi, M. 82
Adachi, S. 415
Aihara, K. 78, 82, 86, 90, 98
Aiyoshi, E. 130, 150
Akama, S. 626
Akiyama, N. 759
Akizuki, K. 602
An, S.-K. 179
Ando, H. 682
Aoki, T. 291
Arai, F. 512
Arai, K. 150
Aramaki, S. 266
Arendt, J.A. I-9
Asai, T. 305, 309
Asama, H. 34, 50
Ascott, R. 12
Asharif, M.R. 403

B

Baba, T. 28
Bae, J.-I. 170
Baek, S.-M. 658
Belew, R.K. OP-22
Bubnicki, Z OP-34

C

Casti, J.L. OP-15
Chakarborty, I. 715
Chang, M. 536
Cho, D.-Y. 236
Cho, S.-B. 297
Choi, J. 448
Choi, J.W. 188
Chung, M-J 650
Colgate, S. OP-21

D

Dai, R. I-27
Das, S. 150

E

Eggenberger, P. 324
Endo, I. 34
Endo, S 391, 395, 407
Eom, T.-D. 228, 763
Etani, T. 16

F

Fei, Y. I-27
Filliat, D. 745
Fredslund, J. I-9
Fujii, N. 166
Fujii, T. 34
Fujinaka, T. 146
Fujisawa, K. 291
Fujiwara, Y. 61, 65, 250,
Fukuda, T. 504, 512
Furuhashi, T. 508

G

Gao, D.-Z. 662
Garis, H.D. 606, 610
Gen, M. 448, 452, 456, 460
Gers, F. 606, 610
Goto, S. 630
Goto, Y. 200, 204, 208, 283
Guo, Q. 274

H

Hagan, S. 480, 588
Hagiwara, M. 138
Han, M.-C. 175
Han, S.-H. 170, 188
Han, S.-J. 179
Hanazaki, I. 602
Harada, H. 375

Hart, J. I-1
Haruyama, M. 122
Hashem M.M.A. 516, 618
Hashimoto, H. 28, 40
Hashimoto, S. 504
Hatanaka, T. 301
Hatano, I. 166
Hattori, K. 362
Hayakawa, S. 291
Hayashi, H. 90
Hidai, K. 283
Hirafuji, M. 480, 588
Hirasawa, K. 694, 699
Hirashima, Y. 118
Hirata, Y. 50
Hirayama, H. 126, 733
Hirayama, T. 358
Hiroi, R. 560
Hokari, S. 344
Hong, S.-G. 763
Honma, N. 741
Hori, G. 86
Hori, K. 464
Horie, R. 130
Horio, Y. 90
Hough, M. 606, 610
Hu, D. I-31, 472
Hu, J. 694
Hu, Z. 46
Huang, J. 138
Hwang, G.-H. 183

I

Ichikawa, A. 719
Ichiki, T. 576
Ida, K. 448, 456, 460
Ikeguchi, T. 94
Imai, M. 16
Inaba, M. 737
Inabayashi, S. 636
Inabayashi, S. 707
Inoue, A 118
Ishida, Y. 69, 419
Ishiguro, A. 324
Ishii, H. 682
Ishinishi, M. 524
Islam, M.M. 305, 309
Isogai, M. 512
Itiki, C. 313, 321
Itoh, K. 719
Ivan, L. 480
Iwakura, H. 544
Iwamoto, T. 58
Izumi, K. 516, 520, 618, 626

J

James, D.J.G. I-2
Jiang, R. I-31, 472
Jinnai, H. 568
Jo, Y.-G. 220
Jun, H.-B. 232
Jung, M.-J. 670

K

Kaetsu, H. 34, 50
Kakazu, Y. 350
Kalaba, R. 313, 321
Kamano, T. 614
Kambara, T. 362, 468, 644
Kang, H. 216, 220
Kashimori, Y. 362, 468, 644
Katai, O. 674
Kataoka, Y. 614
Kato, Y. 636
Kawabata, K. 50
Kawaji, S. 46
Kawakami, K. 154
Kawata, S. 122
Keller, E.L. 150
Kim, D.-Y. 650
Kim, H.-S. 670
Kim, J.-H. 670
Kim, J.R. 460
Kim, M.-S. 763
Kim, S. 192
Kim, S.-I. 576, 584
Kimura, H. OP-4
Kimura, K. 371
Kimura, T. 196
Kinjo, H. 431, 435

Kinoshita, M. 246
Kinouchi, Y. 707
Kita, H. 330, 334
Kitamura, S. 162, 287
Kitazawa, M. 371, 375
Kitazoe, T. 568, 576, 584
Kiyooka, T. 383
Kizu, S. 415
Kobayashi, I. 90
Kobayashi, S. 334
Kodama, S. 114
Kodjabachian, J. 745
Koga, H. 556
Kohashi, T. 340
Kohata, N. 28
Kojima, F. 504
Kojima, M. 751
Konar, A. 690, 715
Kondo, H. 711
Kondo, T. 324
Korkin, M. 606, 610
Kosaka, T. 142
Kosuge, K. 50
Kryssanov, V.V. 162
Kubo, M. 751
Kubota, N 504
Kuc, T.-Y. 658
Kumagai, S. 106
Kunii, Y. 40
Kuribayashi, D. 34
Kuribayashi, K. 440, 444
Kurono, S. 266

L

Lachner, K. OP-21
Lee, C.-Y. 763
Lee, D.-W. 224, 232
Lee, D.Y. 188
Lee, I.-J. 658
Lee, J.-J. I-16, 228, 279, 763
Lee, J.-M. 179, 188
Lee, M.C. 188
Lee, M.H. 170, 175, 179, 183, 188, 192
Lee, S.-Y. 699
Lehnert, B. OP-21
Lei, X.-Y. 379
Li, D. 682
Li, F. 755
Li, S.-R. 755
Li, Y.-Z. 448, 456
Liang, X. 666
Liu, Z.-J. 767
Lu, B. 694
Lu, Q. 755
Luisi, L. OP-21
Lund, H.H. I-9
Luo, Y. I-31, 472

M

Maekawa, S. 61, 65
Maeshiro, T. 258
Manabe, K. 572, 580
Mandal, A.K. 690, 715
Mautner, C. OP-22
Masuda, S. 118
Matrosov, V.M. 399
Matrossov, I.V. 411
Matsumoto, G. OP-7
Mayer, B. OP-21
Meyer, J.-A. 745
Mignonneau, L. 73
Mikami, S. 751
Min, S.-K. 216
Minamitani, H. 134
Mishima, T. 200, 204, 208, 212, 283
Miyagi, H. 395, 407
Miyagi, H. 439
Miyake, T. 379
Miyamoto, T. 106
Miyamoto, Y. 266
Miyano, T. 98
Mizoguchi, H. 200, 204, 208, 212 283
Mizuhara, H. 363
Mizutani, M. 707
Mochizuki, M. 134
Mori, N. 330
Mun, K.-J. 183

Munekata, M. 358
Murakami, K. 444
Murakoshi, T. 602
Murao, H. 287
Murase, K. 305, 309
Murata, J. 694
Murata, T. 354

N

Nagata, F. 626
Nakagawa, M. 552, 556, 560
Nakagawa, T. 737
Nakagawa, T. 682
Nakajima, H. 106
Nakajima, R. 196
Nakamura, A. 275
Nakamura, M. 630
Nakanishi, M. 427
Nakano, K. 723
Nakasuka, S. 464
Nakatsu, R. 5
Namatame, A. 524, 528, 532
Nanayakkara, D.T. 520
Narita, Y. 362
Natsuyama, H. 313, 321
Nawa, N. E. 606, 610
Nerome, M. 395
Nian, W. 423
Nicoud, J. D. I-41
Niino, Y. 648
Nishi, R. 439
Nishikawa, Y. 330, 363
Nishimura, T. 488
Nobuki, O. 387
Nordahl, M.G. 598
Nordin, P. 598
Numata, M. 500

O

Ochi, H. 435
Ochi, Y. 644
Odagiri, R. 305, 309
Ogasawara, T. 275
Ohashi, S. 350
Ohkawa, K. 24
Ohkura, K. 154, 536
Ohnishi, K. 344
Ohuchi, A. 54, 340, 354, 358
759
Okamoto, T. 69
Okazaki, K. 423
Okita, Y. 126, 733
Okuma, S. 291
Omatsu, S. 142, 146
Ono, N. 540
Ono, T. 16
Oshima, T. OP-5
Oura, K. 602
Ozaki, T. 102
Ozawa, K. 484

P

Pagliarini, L. I-9
Park, H.-K. 650
Park, J.-H. 183
Park, K.-B. 279
Patnaik, S. 690

Q

Qu, C. 662

R

Rahman, S.B.A 711
Rasmussen, S. OP-21
Ray, T. I-1
Riquimaroux, H. 572, 580
Rui, F. 666
Rumchev, V. I-2
Ryan, C. 480

S

Sakai, Y. 371
Sakamoto, T. 444
Samadi, S. 544
Sanchez, E. 240
Sanefuji, K. 622
Sano, A. 686
Sato, K. 626
Satou, A. 707
Sawai, H. 415

Sawaragi, T. 674
Sayama, H. 254
Sekine, O. 196
Shi, Z. 102
Shiba, T. 358
Shibata, H. 544
Shibata, T. 20, 24
Shibuta, H. 98
Shigehara, T. 200, 204, 208, 212, 283
Shii, T. 568
Shim, H.-S. 670
Shimoda, H. 682
Shimohara, K. 464
Shinomiya, Y. 196
Shimoyama, Y. 532
Shin, W.-J. 699
Sim K.-B. 224, 232
Sohn, K.-O. 658
Sommerer. C. 73
Son, K. 188
Song, G.-B. 297
Solem, J. OP-21
Solomides, A. 5
Stilman, B. I-35
Suehiro, T. 275
Sugawara, K. 488, 492, 496, 500, 737
Sugi, T. 630
Sugisaka, M. I-22, 648, 654, 729, 767
Suzuki, H. 118
Suzuki, K. 54, 340, 354, 358 759
Suzuki, T. 614
Suzuki, T. 291
Suzuki, Y. 262
Svinin, M. M. 158
Shouji, F. 707

T

Tabuchi, F. 431
Tabuse, M. 548
Takadama, K. 464
Takahashi, O. 334
Takahide, M. 28
Takai, S. 110
Takayanagi, T. 340
Takeda, A. 496
Takeda, H. 741
Takeo, K. 50
Takeuchi, I. 508
Takiya, S. 358
Takuno, M. 301
Tamaki, H. 162
Tamura, S. 423, 636
Tamura, S. 301
Tamura, Y. 102
Tanaka, A. 114
Tanaka, H. 244, 262
Tanaka-Yamawaki, M. 548
Tang, J. 440, 622
Tanie, K. 20, 24
Taura. T. 592
Taylor, C. OP-1
Terano, T. 464
Teshiba, M.W. 204, 212
Tilden, M. OP-2
Toma, N. 391
Toma, T. 403
Tomizuka, M. 723
Tosa, N. 1, 5
Toyoda, S. 640
Tsuchiya, K. 320
Tsuji, T. 279
Tsujita, K. 320
Tsukune, H. 275
Tsutsumi, K. 358

U

Uchikawa,Y. 324
Uchimura, K. 46
Uchiyama, H 564
Ueda, K. 154, 158, 166, 536
Uezato, E. 431
Uno, K. 528
Uosaki, K. 301
Uribe, A.P. 240
Ushio, T. 114

V

Vadakkepat, P. 670

W

Wada, M. 246, 751
Wakamatsu, H. 703
Wang, X. 654
Wang, X. 729
Watanabe, K. 516, 520, 618, 622, 626
Watanabe, M. 78
Wei, Y. 305, 309
Wells, W. R. OP-28
Whitten, D. OP-21
Wu, J.-L. 363, 367, 371, 375, 379, 383, 387
Wu, W. 682
Wu, X. I-22

X

Xi, H. 472
Xu, X. 666

Y

Yamada, K. 391, 395, 407
Yamada, K. 158
Yamada, S. 484, 496, 500
Yamaguchi, A. 751
Yamaguchi, T. 28
Yamamoto, H. 270
Yamamoto, M. 354, 358, 759
Yamamoto, T. 431, 435
Yamashita, T. 54
Yamauchi, S. 407
Yanagawa, H. 344
Yasuno, T. 614
Yim, W. OP-28
Yoneyama, J. 719
Yonezu, M. 427
Yoshida, H. 367
Yoshida, S. 540
Yoshida, T. 427
Yoshida, Y. 614
Yoshihara, I. 484, 488, 492, 496, 500, 737
Yoshii, S. 350
Yoshikawa, H. 682
Yoshimi, T. 592
Yoshioka, M. 146

Z

Zhang, B.-T. 236
Zhang, C. 662
Zhang, X. 703
Zhang, Y.-G. I-22
Zheng, M.-H. 468
Zhou, G. 452