PROCEEDINGS OF THE SIXTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 6th '01)

Vol.2

Jan. 15-Jan. 17, 2001
U-Port, Gotanda, Tokyo, JAPAN

Editors : Masanori Sugisaka and Hiroshi Tanaka
ISBN4-9900462-1-8

Proceedings of The Sixth International Symposium on

ARTIFICIAL LIFE AND ROBOTICS

(AROB 6th '01)

for Cognitive and Behavioral Intelligent Artificial Liferobot

January 15-17, 2001
U-Port, Gotanda, Tokyo, JAPAN

Editors: Masanori Sugisaka and Hiroshi Tanaka

THE SIXTH INTERNATIONAL SYMPOSIUM
ON
ARTIFICIAL LIFE AND ROBOTICS
(AROB 6th '01)

ORGANIZED BY

Organizing Committee of International Symposium on Artificial Life and Robotics(AROB)

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

The Institute of Electronics, Information and Communication Engineers (IEICE, Japan)
The Institute of System, Control and Information Engineers (ISCIE, Japan)

SUPPORTED BY

Asahi shimbun
Beppu City Hall
Kanto Bureau of Economy, Trade and Industry (to be expected)
Kyushu Bureau of Economy, Trade and Industry
NHK Oita Station
Nihon Keizai Shimbun, INC.
THE NIKKAN KOGYO SHINBUN, LTD
OBS Broadcast Company
Oita Asahi Broadcasting
Oita Municipal Government
Oita Prefectural Government
Oitagodo Shinbunsya
PRESS THE NISHINPPON
Science and Technology Agency
TOS Broadcast Company
The Mainichi Newspapers
THE YOMIURI SHIMBUN

HONORARY PRESIDENT

M. Hiramatsu (Governor, Oita Prefecture)

ADVISORY CCOMMITTEE CHAIRMAN

F. Harashima (President, Tokyo Metropolitan Institute of Technology, Japan)

GENERAL CHAIRMAN

Masanori Sugisaka (Oita University)

CO-GENERAL CHAIRMAN (PROGRAM)

Hiroshi Tanaka (Tokyo Medical & Dental University)

CO-CHAIRMAN

John Casti (Santa Fe Institute)

ADVISORY COMMITTEE

S.Fujimura (The University of Tokyo, Japan)
F.Harashima (President, Tokyo Metropolitan Institute of Technology, Japan)(Chairman)
H.Kimura (The University of Tokyo, Japan)
S.Ueno (Kyoto University, Japan)

INTERNATIONAL ORGANIZING COMMITTEE

W.B.Arthur (Santa Fe Institute, USA)
Z.Bubnicki (Wroclaw University of Technology, Poland)
W.Banzhaf (University of Dortmund, Germany)
C.Barrett (Los Alamos National Laboratory, USA)
J.L.Casti (Santa Fe Institute, USA)
J.P.Crutchfield (Santa Fe Institute, USA)
J.M.Epstein (Santa Fe Institute, USA)
T.Fukuda (Nagoya University, JAPAN)
D.J.G.James (Coventry University, UK)
S.Kauffman (Santa Fe Institute, USA)
C.G.Langton (Santa Fe Institute, USA)
J.J.Lee (KAIST, Korea)
Y.Li (Tsinghua University, China)
R.G.Palmer (Santa Fe Institute, USA)
S.Rasmussen (Santa Fe Institute, USA)
T.S.Ray (University of Oklahoma, USA)
P.Schuster (Santa Fe Institute, USA)
M.Sugisaka (Oita University, Japan) (Chairman)
H.Tanaka (Tokyo Medical & Dental University, Japan)
C.Taylor (University of California-Los Angeles, USA)
W.R.Wells (University of Nevada-Las Vegas, USA)
Y.G.Zhang (Academia Sinica, China)

INTERNATIONAL STEERING COMMITTEE

M.Bedau (Reed College, USA)
Z.Bubnicki (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 (Coventrry University, UK)
J.J.Lee (KAIST, Korea)
G.Matsumoto (RIKEN, Japan)
M.Nakamura (Saga University, Japan)
S.Rasumussen (Santa Fe Institute, USA)
T.S.Ray (University of Oklahoma, USA)
M.Sugisaka (Oita University, Japan) (Chairman)
H.Tanaka (Tokyo Medical & Dental University, Japan)
C.Taylor (University of California-Los Angeles, USA)
K.Tsuchiya (Kyoto University, Japan)
W.R.Wells (University of Nevada-Las Vegas, USA)
Y.G.Zhang (Academia Sinica, China)

INTERNATIONAL PROGRAM COMMITTEE

K.Abe (Tohoku University, Japan)
K.Aihara (The University of Tokyo, Japan) (Co-chairman)
M.Bedau (Reed College, USA)
R.Belew (University of California- San Diego, USA)
Z.Bubnicki (Wroclaw University of Technology, Poland)
T.Christaller (CMD-German National Research Center for Information Technology, Germany)
T.Fujii (RIKEN, Japan)
M.Gen (Ashikaga Institute of Technology, Japan)
T.Gomi (AAI, Canada)
I.Harvey (University of Sussex, UK)
H.Hashimoto (The University of Tokyo, Japan) (Co-chairman)
H.Hirayama(Asahikawa Medical College, Japan)
P.Husbands (University of Sussex, UK)
K.Ito (Tokyo Institute of Technology, Japan)
J.Johnson (The Open University, UK)
Y.Kakazu (Hokkaidou University, Japan)
R.E.Kalaba (University of Southern California, USA)
H.Kashiwagi (Kumamoto University, Japan)
O.Katai (Kyoto 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)
S.Kumagai (Osaka University, Japan)
Carl G. Looney (University of Nevada-Reno)
H.H.Lund (University of Aarhus, Denmark)

M.Nakamura (Saga University, Japan)
R.Nakatsu (Santa Fe Institute, Japan)
H.H.Natsuyama (Advanced Industrial Materials, USA)
S.Omatsu (University of Osaka Prefecture, Japan)
T.Omori (Tokyo University of Agriculture &Technology, Japan)
R.Pfeifer (University of Zurich-Irchel, SwitZerland)
T.S.Ray (University of Oklahoma, USA) (Co- chairman)
Y.Sankai (University of Tsukuba, Japan)
T.Sawaragi (Kyoto University, Japan)
T.Shibata (MITI, MEL, Japan)
K.Shimohara (ATR, Japan)
L.Steels (VUBAI Laboratory, Belgium)
M.Sugisaka (Oita University, Japan)
S.Tamura (Osaka University, Japan)
H.Tanaka (Tokyo Medical & Dental University, Japan) (Chairman)
N.Tosa (ATR, Japan)
K.Ueda (Kobe University, Japan)
A.P.Wang (Arizona State University, USA)
K.Watanabe (Saga University, Japan)
X.Yao (The University of New South Wales, Australia)
W.R.Zimmer(GMD-Japan Research Laboratory, Japan)

LOCAL ARRANGEMENT COMMITTEE

K.Nakano (University of Electro-Communications, Japan)
K.Okazaki (Fukui University, Japan)
S.Sato (Director, Research and Development Center, Oita University, Japan)
K.Shibata (Tokyo Institute of Technology, Japan)
K.Shigemitsu (Oita Industrial Research Institute, Japan)
M.Sugisaka (Oita University, Japan)
Y.Suzuki (Tokyo Medical&Dental University)
H.Tsukune (Director, Oita Industrial Research Institute, Japan)
X.Wang (Oita Institute of Technology, Japan)
I.Yoshihara (Miyazaki University, Japan)

TOPICS

Hardware Oriented Topics are welcome
in the fields given by

Artificial Brain Research
Artificial Intelligence
Artificial Life
Artificial Liferobotics
Artificial Living
Artificial Mind Research
Bioinformaitcs
Brain Science
Chaos
Cognitive Science

Complexity
Computer Graphics
Evolutionary Computations
Fuzzy Control
Genetic Algorithms
Human-Machine Cooperative Systems
Human-Welfare Robotics
Innovative Computations
Intelligent Control and Modeling
Micromachines
Micro-Robot World Cup Soccer Tournament
Mobile Vehicles
Molecular Biology
Neural Networks
Neurocomputers
Neurocomputing Technologies and Their Applications for Hardware Robotics
Robust Virtual Engineering
Virtual Reality

COPYRIGHTS

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

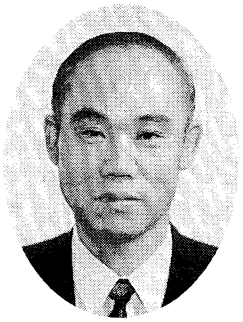
All correspondence relating to the symposium should be addressed to:

Prof. Masanori Sugisaka
General Chairman of International Symposium
On Artificial Life and Robotics
(AROB)
AROB Secretariat
Dept. of Electrical and Electronic Engineering
Oita University
700 Dannoharu, Oita 870-1192
JAPAN
TEL 001-81-97-554-7831
FAX 001-81-97-554-7841
E-MAIL msugi@cc.oita-u.ac.jp
arobsecr@oita-cc.cc.oita-u.ac.jp

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

MESSAGE

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



It is my great honor to invite you all to the upcoming International Symposium on Artificial Life and Robotics. The first symposium was held in February (18-20) 1996, B-Con Plaza, Beppu, Oita, Japan. That symposium was organized by Oita University under the sponsorship of the Japanese Ministry of Education, Science, Sports, and Culture (Monbusho), and co-sponsored by Santa Fe Institute (USA), SICE, RSJ, and IEEJ, (Japan). This symposium invites you to discuss the development of new technologies in the 21st century, concerning Artificial Life and Robotics, based on simulation and hardware.

I would like to express my sincere thanks to the Science and International Affairs Bureau, Monbusho, Japanese Government, for their repeated support.

We hope that AROB will facilitate the establishment of an international joint research institute on Artificial Life and Robotics. I hope that you will obtain fruitful results from the exchange of ideas during the symposium.

Masanori Sugisaka
M. Sugisaka

January 5, 2001

MESSAGE

John L. Casti
Vice Chairman of AROB
(Professor, Santa Fe Institute, USA)

For the past 300 years or more, science has focused on understanding the material structure of systems. This has been evidenced by the primacy of physics as the science par excellence, with its concern for what things are made of. The most basic fact about science in the 21st century will be the replacement of matter by information. What this means is that the central focus will shift from the material composition of systems—what they are—to their functional characteristics—what they do. The ascendancy of fields like artificial intelligence, cognitive science, and now artificial life are just tips of this iceberg.

But to create scientific theories of the functional/informational structure of a system requires employment of a totally different type of laboratory than one filled with retorts, test tubes or bunsen burners. Rather than these labs and their equipment designed to probe the material structure of objects, we now require laboratories that allow us to study the way components of systems are connected, what happens when we add/subtract connections, and in general, experiment with how individual agents interact to create emergent, global behavioral patterns.

Not only are these "information labs" different from their "matter labs" counterparts. There is a further distinction to be made even within the class of information labs. Just as even the most well-equipped chemistry lab will help not one bit in examining the material structure of, say, a frog or a proton, a would-be world designed to explore traders in a financial market will shed little, if any, light on molecular evolution.

Since the very first Artificial Life meeting in 1987 in Los Alamos, New Mexico, the Santa Fe Institute (SFI) has been at the forefront of this shift in emphasis from matter to information. By the same token, SFI has actively supported such research initiatives in every corner of the world. This support has extended to the Artificial Life and Robotics meetings here in Japan, since the time of the very first meeting in 1996. Each year, researchers from the SFI faculty have come to Japan to meet with others at these AROB meetings, in order to present edge-of-the-frontier ideas and to exchange views on how the fields of ALife and robotics are progressing. So it is a great pleasure for me to again represent SFI on the Organizing Committee of AROB6, and to welcome everyone to this event.

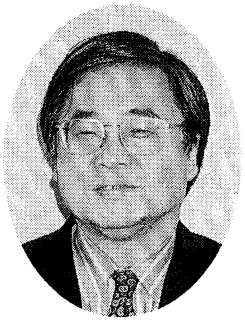
John CASTI

J.L. Casti

January 5, 2001

MESSAGE

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 Sixth International Symposium on Artificial Life and Robotics (AROB 6th '01). 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, 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 or Alife approach now attracts wide interests as a new paradigm of science and engineering. Take an example in the field of bioscience. As is well known by the name of HGP (Human Genome Project), vast amount of genome information brings about not only from human genome but also various species like several bacterias, yeast, warm, fly. However, as a plenty of genome data becomes available, it becomes sincerely recognized that the framework by which these genome data can be understood to make a whole picture of life is critically needed. The complex systems or Alife approach is now actually expected to be an efficient methodology to integrate this vast amount of data.

This example shows the complex system approach is very promising and becomes widely accepted as a paradigm of next generation of science and engineering. 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 Tokyo.

H. Tanaka

January 5, 2001

TECHNICAL PAPER INDEX

Plenary Lecture (Plenary Talks)

P1 *Evolutionary Computing in Robotics*

T. Fukuda (Nagoya University, Japan) ·····offprint

P2 *E-CELL Project: Building Working Cells in Silico*

M. Tomita (Keio University, Japan) ·····offprint

MA1: Intelligent Control and Modeling I (General Session)(Room Sakura)

MA1-1 *Adaptive fuzzy controller for a class of uncertain operation systems* ····· 1

L. Siwek (Wroclaw University of Technology, Poland)

MA1-2 *Design and optimization of fuzzy controllers based on the operator's knowledge* ····· 5

H. Bae, S. Kim, M.H. Lee (Pusan National University, Korea)

MA1-3 *Fuzzy identification of chaotic and complex behavior of human operator stabilizing an inverted pendulum on a cart* ····· 9

Y. Kawazoe (Saitama Institute of Technology, Japan)

MA1-4 *Adaptive visual control of a sm5 robot with a hand-eye coordinated camera* ·····13

H. D. Kim (Yenam Collage, Korea)
D. Y. Jeong, S. H. Han (Kyungnam University, Korea)
M. H. Lee (Pusan National University, Korea)
H. Hashimoto (University of Tokyo, Japan)

MA2: Brain Science and Complexity I (General Session)(Room Sakura)

MA2-1 *Dynamical behaviors of extended Hogg-Huberman model* ·····19

T. Tanaka, J. Shibata, K. Okuhara (Hiroshima Prefecture University, Japan)
M. Inoue (Kagoshima University, Japan)

MA2-2 *A chaos oscillator model of temporal-frequency characteristics on human visual search* ……23

H. Mizuhara, T. Saito (Yamaguchi University, Japan)
J. L. Wu (Kagawa University, Japan)

MA2-3 *Analysis of small-world networks with high broadcast speed* ……27

T. Maeshiro, K. Shimohara (ATR, Japan)
N. Ohi (Ryukoku University, Japan)

MA3: Artificial Life I (General Session)(Room Sakura)

MA3-1 *On multistage breeding processes* ……31

T. Odanaka, T. Arimizu (Tokyo Metropolitan Institute of Technology, Japan)

MA3-2 *Sequential dynamical systems* ……34

C. L. Barrett, H. Mortveit, C. M. Reidys (Los Alamos National Laboratory, USA)

MA3-3 *A computational model of the interaction between environmental dynamics and economic behaviors* ……37

T. Yamaoka, T. Arita (Nagoya University, Japan)

MA3-4 *The problems under the dynamic environments need the consideration of the evaluation standard between the environments that are changing.* ……offprint

K. Yamasaki (Tokyo University of Information Science, Japan)

MA3-5 *Stochastic Matching Agents* ……41

G. Hernandez, F. Nino, Y. Garcia (National University of Columbia, Columbia)
K. Khouri (Northwest Mississippi Community College, USA)

MB1:The Significance and Meaning for Emergence (Organized Session)(Room Kaede)

MB1-2 *The third wholeness driving emergent property* ……offprint

Y. G. Pegio (University of Kobe, Janpan)

MB1-6 *The physical constraint programming for real emergent system* ·····offprint

K Hajiri (IBM Tokyo Research Laboratory, Japan)

MB2: Communications with Life-like Creatures and Robots (Organized Session) (Room Kaede)

MB2-1 *What is the ultimate form of communications ?* ·····45

R. Nakatsu (ATR, Japan)

MB2-2 *Muu: Embodied interface for social bonding* ·····offprint

S. Shoji, N. Suzuki, M. Okada (ATR, Japan)

MB2-3 *Robovie: Communication technologies for a Social Robot* ·····50

M. Imai, T. Ono, H. Ishiguro (ATR, Japan)

MB2-4 *Mixing with Aliens: Life and music on Gakki-mon Planet* ·····offprint

R. Berry, P. Dahlstadt, C. Haw (ATR, Japan)

MB3: Neural Network, Recognition and Behavior Control (Organized Session) (Room Kaede)

MB3-1 *Integration system for emotional recognition using visible and infrared face images and voice* ·····54

T. Ikeda, Y. Yoshitomi, T. Kitazoe, S. I. Kim (Miyazaki University, Japan)

MB3-2 *Fractal evaluations of fish school movements in simulations and real observations* ·····58

T. Shinchi, T. Kitazoe, M. Tabuse (Miyazaki University, Japan)
H. Nishimura (Hyogo University, Japan)
N. Azuma (Hirosaki University, Japan)

MB3-3 *Application of visual recognition neural network to hearing system for continuous speech* ·····62

M. Funamori, T. Kitazoe, T. Ichiki (Miyazaki University, Japan)

MB3-4 *Evolutionary robot controllers with competitive and cooperative neural network* ……66

A. Todaka, T. Kitazoe, M. Tabuse, K. Sugihara (Miyazaki University, Japan)

MB3-5 *Control systems for real robot using classifier systems* ……70

T. Jinguuji, M. Tabuse, K. Sugihara (Miyazaki University, Japan)

MC1: Genetic Algorithms and Evolutionary Computation I (General Session) (Room Natsume)

MC1-1 *Evolutionary simulations of "SUMIWAKE" habitat segregation in a finite and heterogeneous ecosystem* ……73

T. Oohashi, T. Maekawa, (ATR, Japan)
O. Ueno, (Gifu University School of Medicine, Japan)
N. Kawai, (Foundation for Advancement of International Science, Japan)
E. Nishina, (National Institute of Multimedia Education, Japan)

※MC1-2 *A macro-micro evolutionary algorithm: multi agents model for optimization* ……77

S. K. Oh, K. H. Seo, J. J. Lee (Korea Advanced Institute of Sience and Technology, Korea)

※This paper would be changed from oral session to poster session.

MC1-3 *DNA computing for shortest path problem* ……81

N. Matsuura, M. Yamamoto, T. Shiba, A. Ohuchi (Hokkaido University, Japan)

MC1-4 *The optimization of neural network structure using genetic algorithm* ……85

M. Itou, M. Sugisaka (Oita University, Japan)

MC2: Neural Networks I (General Session)(Room Natsume)

MC2-1 *Self-learning probabilistic neural network hardware using reconfiguable LSIs* ……89

N. Aibe, M. Yasunaga (University of Tsukuba, Japan)
I. Yoshihara (Miyazaki University, Japan)

MC2-2 *New camera calibration method for stereo vision using neural network* ····· 93

Y. Kang, Y. Shin (Kwangju University, Korea)
N. Kim (Chosun College of Science & Technology, Korea)

MC2-3 *Structure Minimization using impact factor in neural networks* ····· 97

K. H. Seo, J. S. Song, J. J. Lee (KAIST, Korea)

MC2-4 *Dynamical associative memory model consists of structurally unstable oscillators and learning rule based on the internal state of the network* ·····101

K. Kojima, K. Ito (Tokyo Institute of Technology, Japan)

MC3: Robotics I (General Session)(Room Natsume)

MC3-1 *Network-based human assist robotic system using CORBA* ·····105
S. Jia, K. Takase (University of Electro-Communications, Japan)

MC3-2 *Cooperative behavior of multi robot system with simple interaction* ·····109

K. Sugawara, (University of Electro-Communications, Japan)
M.Sano, (University of Tokyo, Japan)
I. Yoshihara, (Miyazaki University, Japan)
K. Abe, (Tohoku University, Japan)
T. Watanabe, (University of Electro-Communications, Japan)

MC3-3 *Resolved motion rate control of a free-floating unsderwater robot with horizontal planar 2 link manipulator* ·····113

S. Sagara, T. Danjoh, M. Tamura, R. Katoh (Kyushu Institute of Technology, Japan)

MC3-4 *Legged robot control using multi-agent robot language* ·····117

M. Obayashi, H. Nishiyama, F. Mizoguchi (Science University of Tokyo, Japan)

MC3-5 *Control of manipulator mounted on floating underwater robot while maintaining vehicle attitude* ·····121

M. Tamura, S. Sagara, R. Katoh (Kyushu Institute of Technology, Japan)

TB-1: Softcomputing in Robotic Intelligence (Organized Session)(Room Kaede)

TB-1-1 *Achieving synergy through acquisition of human skill* ·····126

T. Nanayakkara, K. Watanabe, K. Kiguchi, K. Izumi (Saga University, Japan)

TB-1-2 *On the fuzzy model-based control for an acrobot* ·····130

K. Watanabe, K. Izumi, K. Kiguchi (Saga University, Japan)

TB-1-3 *Control for a rings gymnastic robot using fuzzy reasoning and genetic algorithms* ·····134

T. Yamada, Keigo Watanabe, K. Kiguchi, K. Izumi (Saga University, Japan)

TB-1-4 *Fuzzy behavior-based control of mobile robot in dynamic environments using modules learned in static environments* ·····138

K. Izumi, K. Watanabe, K. Kiguchi (Saga University, Japan)

TB-1-5 *Energy optimal gait analysis of quadruped robots* ·····142

K. Kiguchi, Y. Kusumoto, K. Watanabe, K. Izumi (Saga University, Japan)
T. Fukuda (Nagoya University, Japan)

TB-2: Artificial Brain (Organized Session)(Room Kaede)

TB-2-1 *MemeStorms: cellular working memory and dynamics of judgment* ·····146

A. Buller, T. Chodakowski , K. Shimohara (ATR, Japan)
L. Kaiser (Wroclaw University, Poland)
A. Nowak (Warsaw University, Poland)

TB-2-2 *Dynamic fuzzy sets for cognitive modeling* ·····150

A. Buller (ATR, Japan)

TB-2-3 *Suitable evolutionary strategies for large scale neural networks* ·····152

P. Eggenberger (ATR, Japan)

TB-2-4 *Development of an artificial brain structure for the behavior control of the welfare liferobot* ·····156

A. Loukianov, M. Sugisaka (Oita University, Japan)

TB-3: Methodology of Emergent Synthesis (Organized Session)(Room Kaede)

TB3-1 *Methodology of emergent synthesis* ……160

K. Ueda, H. Tamaki, I. Hatono (Kobe University, Japan)

TB3-2 *A coevolutionary approach to adaptive encoding for genetic algorithms* ……164

H. Murao, A. Yamamoto, H. Tamaki, S. Kitamura (Kobe University, Japan)

TB3-3 *Adaptive segmentation of the state space based on bayesian discrimination* in reinforcement learning ……168

K. Yamada, K. Ohkura, K. Ueda (Kobe University, Japan)
M. Svinin (RIKEN, Japan)

TB3-4 *A Study on the multicriteria optimization support by using evolutionary algorighms* ……172

T. Inamoto, K. V. Victor, H. Tamaki, S. Kitamura (Kobe University, Japan)

TB3-5 *An evolutionary approach to decentralized reinforcement learning for walking robots* ……176

S. Ushio, K. Ueda (Kobe University, Japan)
M. Svinin, S. Hosoe (RIKEN, Japan)

TB3-6 *Emergence of supply chains by producers' selection of suppliers* ……180

D. Nakanishi, I. Hatono, K. Ueda (Kobe University, Japan)

TC1: Brain Science and Complexity II (General Session)(Room Natsume)

TC1-1 *Chaotic information maximization for blind source separation* ……184

W. Yu, H. Yokoi, Y. Kakazu (Hokkaido University. Japan)

TC1-2 *An experimental platform for analyzing interaction between human, machine and environment* ……188

K. Morikawa (Matsushita Electric Industrial Co., Ltd., Japan)
N. Oka (Matsushita Research Institute Tokyo, Inc., Japan)
S. Agarwal (University of California, San Diego)

TC1-3 *Criticality of cooperative society* ·····192

M. Kubo, H. Satoh, A. Namatame (National Defense Academy, Japan)

TC2: Intelligent Control and Modeling II (General Session)(Room Natsume)

TC2-1 *Real time gaze control of active head-eye system without calibration* ·····196

D. Y. Kim, J. R. Ryoo, H. K. Park, M. J. Chung
(KAIST, Korea)

TC2-2 *Fast and stable learning in direct-vision-based reinforcement learning* ·····200

K. Shibata, M. Sugisaka (Oita University, Japan)
K. Ito (Tokyo Institute of Technology, Japan)

TC2-3 *Evolving neurofuzzy systems for system identification* ·····204

Y. Chen, S. Kawaji (Kumamoto University, Japan)

TC2-4 *Labeling Q-learning in hidden state environments* ·····208

H. Y. Lee, K. Abe (Tohoku University, Japan)
H. Kamaya, (Hachinohe National College of Technology, Japan)

TC3: Artificial Life II (General Session)(Room Natsume)

TC3-1 *The voice sound source separation based on the grouping in the frequency domain* ·····212

K. Ninagawa, T. Umeyama, Y. Sagawa, N. Sugie (Meijo University, Japan)

TC3-2 *Social evolution in imperfect world* ·····216

H. Satoh, K. Uno, M. Kubo, A. Namatame (National Defense Academy, Japan)

TC3-3 *Application of uncertain variables and logics to complex intelligent systems* ·····220

Z. Bubnicki (Wroclaw University of Technology, Poland)

TD1: Mobiles Vehicles I (General Session)(Room Aoi)

TD1-1 *Simulation of aerodynamics of micro air vehicles TH380 airfoils* ……224

H. Wu, Z. Zhou, S. Xiong, X. Wang, G. Bao, S. Li, Z. Li
(Tsinghua University, China)

TD1-2 *A study on the optimal design of a mobile robot based upon mobility and Recurrency* ……228

T. S. Jin , J. M. Lee (Pusan National University, Korea)

TD1-3 *Autonomous robot navigation and dynamical system tasks architecture* ……232

M. K. Habib (GMD, Japan)

TD1-4 *Supervised learning technique for a mobile robot controller in a visual line tracking task* ……238

A. Loukianov, M. Sugisaka (Oita University, Japan)

TD2: Related fields I (General Session)(Room Aoi)

TD2-1 *A resolution of the puzzle of the posi-nega switch mechanism in the globally coupled map lattice* ……242

T.Shimada, S.Tsukada (Meiji University, Japan)

TD2-2 *Classification and function prediction of the protein by using data compression* ……246

K. Sugawara, T. Watanabe (University of Electro-Communications, Japan)

TD2-3 *Spatial frequency specific visual adaptation may cause paradoxical transient improvements in visual acuity - experimental and simulation studies -* ……250

Y. Nagai, T. Tanaka, N. Sugie (Meijo University, Japan)
K. Uchida, K. Ueda, H. Onodera (Nagoya Electric Works. Co., Ltd., Japan)

TD2-4 *Teleoperation of CAN-based systems using a Java in the internet* ……254

J.W. Park, M. S. Jeong, J. M. Lee (Pusan National University, Korea)

TD3: Robotics II (.General Session)(Room Aoi)

TD3-1 *Self-organization of behavior-based world model for autonomous mobile robot* ·····258

S.Ishigaki, M. Ida, O.Katai (Kyoto University, Japan)

TD3-2 *Behavior patterns emerging among mobile robots with a diversity of personalities Cooperating in collection cleaning-up tasks* ·····262

E. Uozumi, Y. Sagawa, N. Sugie (Meijo University, Japan)

TD3-3 *Adaptive and economic data representation in control architectures of autonomous real world robots* ·····266

J.Fischer, R. Breithaupt, (GMD, Germany)
M. Bode (Westfaehlische Wilhelms University, Germany)

TD3-4 *Implementation of a virtual manufacturing line for agile assembly* ·····272

J. Y. Choi, S. M. Cha, S. H. Kim, J. I. Bae, M. H. Lee (Pusan National University, Korea)
H. C. Lee (Yangsan College, Korea)

TD3-5 *Statistical analysis of subjective evaluations of mental commit robot* ·····276

T. Shibata, T. Mitsui, K. Wada, L. Yan, K. Tanie (AIST, MITI, Japan)
A. Touda (Sankyo Aluminum Industry Co., Japan)

WA1:Artificial LifeⅢ (General Session)(Room Sakura)

WA1-1*A framework of deliberative decision making in "conscious" software agents* ·····280

R. Kondadadi, S. Franklin (University of Memphis, USA)

WA1-2 *Object recognition using the stereo vision for underwater robots* ·····284

H. Tanaka, E. Shimizu , M. Ito (Tokyo Uniersity of Mercantile Marine, Japan)

WA1-3 *Humanized robot (Hubot) with K-Artificial brain* ·····288

Y. G. Zhang (Academia Sinica, China)
M. Sugisaka (Oita University, Japan)

WA1-4 *Investigation summary of physiological and psychological influence of animals on people* ·····292

K. Wada (University of Tsukuba, Japan)
T. Shibata, T. Mitsui, L. Yan, K. Tanie (AIST, MITI, Japan)

WA2: Mobiles Vehicles II (General Session)(Room Sakura)

WA2-1 *Path planning of a mobile robot by optimization and reinforcement learning* ……296

H. Igarashi (Kinki University, Japan)

WA2-2 *Integrated path planning and steering control with multisensor fusion for intelligent mobile robots* ……301

H. Takai, K. Tachibana (Hiroshima City University, Japan)
G. Yasuda (Nagasaki Institute of Applied Science, Japan)

WA2-3 *Environment Mapping for Khepera Robot: A New Method by Fusion of pseudo Information Measures* ……305

M. R. Asharif (University of the Ryukyus, Japan)
B. Moshiri, R. HoseinNezhad (University of Tehran, Iran)

WA2-4 *The controlling of the welfare robot prototype* ……309

M. Sugisaka, T. Adachi (Oita University, Japan)

WA3: Related fields II (General Session)(Room Sakura)

WA3-1 *Periodic cluster attractors and their stabilities in the turbulent globally coupled map lattice* ……313

T.Shimada, K. Kikuchi (Meiji University, Japan)

WA3-2 Construction of inverse model for data mining by using probabilistic neural *network* ……317

K. Okuhara, H. Fujita, T. Tanaka (Hiroshima Prefecture University, Japan)

WA3-3 *Function discovery system model using non-linear optimization method* ……321

T. Shimomura, K. Yamashita, K. Morita, S. Serikawa(Kyushu Institute of Technology, Japan)

WA3-4 Literary theory on hypertext -enhancing of literature by computer- ……offprint

H. Morita (Siebold University of Nagasaki, Japan)

WA3-5 *Investigation of the tritrophic system of an ecological system by using an abstract chemistry* ····· 325

Y. Suzuki, H. Tanaka (Tokyo Medical and Dental University, Japan)
J. Takabayashi (Kyoto University, Japan)

WA4:Genetic Algorithms II (General Session)(Room Sakura)

WA4-1 *An approach to cooperative genetic algorithms* ····· 329

D. Hu, R. Jiang, Y. Luo (Tsinghua University, China)
K. Y. Szeto (HKUST, China)

WA4-2 *Optimal routing and flow control for multiple I/O data network by using genetic algorithm* ····· 333

K. Okuhara, T. Okada, T. Tanaka (Hiroshima Prefecture University, Japan)

WA4-3 *Emergence of cognitive robot behavior using affordance and genetic algorithm* ····· 337

K. Tagawa, K. Inoue, H. Haneda (Kobe University, Japan)

WA4-4 *A hierarchical parallel distributed genetic algorithm* ····· 341

T. Matsumura, M. Nakamura, S. Tamaki (University of Ryukyus, Japan)
K. Onaga (Okinawa Research Center, Japan)

WA4-5 *A study of Q-learning: dynamic structuring of action space based on genetic algorithm* ····· 345

K. Ito, F. Matsuno (Tokyo Institute of Tecnology, Japan)

WB-1: Biodiversity I (Organized Session)(Room Kaede)

WB-1-1 *Interaction mechanism between DNA and substrates clarified based on molecular dynamics* ····· 349

Y. Komeiji (AIST, Japan)

WB-1-2 *Self-organization in Ecosystem* ····· 353

T. Shimada, S Yukawa, N. Ito (The University of Tokyo, Japan)

WB-1-3 *Cyto-fluid dynamic theory of the origin of base, information, and function* ····· 357

K. Naitoh (Yamagata University, Japan)

WB-1-4 *Complex organics in space to life: a new scenario of chemical evolution* ····· 361

K. Kobayashi, (Yokohama National University, Japan)
H. Yanagawa, (Keio University, Japan)

WB-2: Biodiversity II (Organized Session)(Room Kaede)

WB-2-2 *Fitness landscape of biopolymers and efficient optimization strategy in evolutionary molecular engineering* ····· 365

T. Aita (Tsukuba Research Institute, Japan)
Y. Husimi (Saitama University, Japan)

WB-2-3 *Novel non-metric MDS algorithm with confidence level test* ····· 369

Y. Taguchi (Chuo University, Japan)
Y. Oono (University of Illinois, USA)

WB-2-4 *Optimal design for the evolution of composite mappings* ····· 373

H. Suzuki (ATR, Japan)

WB-3: Non-conventional Computing and Moleware Architecture (Organized Session) (Room Kaede)

WB-3-1 *Computation and life in a reversible cellular space* ····· 377

K. Morita, K. Imai (Hiroshima University, Japan)

WB-3-2 *A design of synchronization algorithms for a large scale of cellular automata* ····· 381

H. Umeo, J. Nishimura, T. Sogabe, M. Maeda (Osaka Electro-Communication University, Japan)

WB-3-3 *Self-timing in biological systems simulated on cellular automata* ····· 387

F. Peper, S. Adachi (Communications Research Laboratory, Japan)
T. Isokawa, N. Matsui (Himeji Institute of Technology, Japan)

WB-3-4 *Optically programmable computations with DNA molecules* ····· 393

D. V. Noort, J. S. McCaskill (GMD, Germany)

WB-3-5 *A novel class of super-turing machines and their robustness analysis* ····· 397

J. Q. Liu, K. Shimohara (ATR, Japan)

WB-4: Biologically Inspired Systems (Organized Session)(Room Kaede)

WB-4-1 *Symbiotic artifact design by bio-informatic and multimodal framework* ····· 401

O.Katai. H. Kawakami, H. Suto (Kyoto University, Japan)
K. Toda (Osaka University, Japan)

WB-4-2 *Reinforcement learning based on resonance between agent and field* ····· 405

H. Yamagishi, H. Kawakami, O.Katai (Kyoto University, Japan)
T. Horiuchi (Osaka University, Japan)

WB-4-3 *Affordance based human motion synthesizing system* ····· 409

H. Ishii, N. Ichiguchi, D. Komaki, H. Shimoda, H. Yoshikawa (Kyoto University, Japan)

WB-4-4 *Interfacing and co-adaptation between interaction loops of human- and machine-autonomies mediated by ecological task environment* ····· 415

T. Sawaragi, Y. Horiguchi (Kyoto University, Japan)

WB-4-5 *Locomotion control of a Multipod robot with CPG principles* ····· 421

K. Tsujita, K. Tsuchiya, S. Aoi, M. Kawakami (Kyoto University, Japan)
A. Onat (Sabanci University, Turkey)

WC1:Molecular Biology I (General Session)(Room Natsume)

WC1-1 *Method for evaluating thermo dynamical and statistical molecular dynamical interactions among the bio molecular particles that are participating for gene expression* ····· 427

H. Hirayama, (Asahikawa Medical College, Japan)
Y. Okita, (Shizuoka University, Japan)

WC1-2 *Distributions of electrical reaction potential, circumferential force and free energy of DNA strands by concentric cylindrical dielectric continua model* ····· 431

H. Hirayama, (Asahikawa Medical College, Japan)
Y. Okita, (Shizuoka University, Japan)

WC1-3 *Using TDGL for estimation of blood heat diffusion* ·····435

Z.Zainon, M. Rizon (University of Malaya, Malaysia)

WC1-4 *Equation for resting membrane potential of cell* ·····439

X.Zhang, H. Wakamatsu (Tokyo Medical and Dental University, Japan)

WC2: Intelligent Control and ModelingⅢ(General Session)(Room Natsume)

WC2-1 *A simple robust tracking controller for robust manipulators using only joint position measurements* ·····443

M. Wada, M. Oya, R. Katoh (Kyushu Institute of Technology, Japan)
H. Honda (YASKAWA ELECTORIC CORPORATION, Japan)

WC2-2 *An optimal control model of a neuromuscular system in human arm movements and its control characteristics* ·····447

T. Kashima (Kobe City College of Technology, Japan)
Y. Isurugi, M. Shima (Hokkaido University, Japan)

WC2-3 *A parametrization of all linear observer for the minimum phase systems* ·····452

K. Yamada (Gunma University, Japan)

WC3: RoboticsⅢ(General Session)(Room Natsume)

WC3-1 *Motion intelligence to use dynamical interfarances and nonlinear friction of mobile manipulators* ·····456

M. Minami, A. Tamamura, T. Asakura (Fukui University, Japan)

WC3-2 *A hierarchical approach for environmental adaptive locomotion control of multi-legged robot* ·····458

T. Odashima (RIKEN, Japan)
Z. W. Luo (Yamagata University, Japan)
S. Hosoe (Nagoya University, Japan)

WC3-3 *A self-organizing visuo-motor map for a redundant manipulator with high manipulability and obstacle free poses* ·····462

N. Okada, E. Kondo (Kyushu University, Japan)
A. Yoshida (Mitsubishi heavy Industries, LTD, Japan)

WC3-4 *Simplified geometric models in planning of skill-based manipulation* ·····466

A. Nakamura, K. Kitagaki (AIST, MITI, Japan)
T. Ogasawara, T. Suehiro (NIST, Japan)

WC4: Neural Networks II (General Session)(Room Natsume)

WC4-1 *Study of universal learning network with branch control* ·····470

Q. Xiong, K. Hirasawa, J. Hu, J. Murata
(Kyushu University, Japan)

WC4-2 *Neural-network controller of articulated robot arm with interference for high-precision contour control* ·····474

T. Zhang, M. Nakamura, S. Goto (Saga University, Japan)
N. Kyura (Kinki University, Japan)

WC4-3 *Chaos control by a stochastic analysis on recurrent neural networks* ·····478

M. Sakai, N. Honma, K. Abe (Tohoku University, Japan)

WC4-4 *Speech recognition using neural networks* ·····482

P. N. Girija, P. S. Rao (Unversity of Hyderabad, India)

WC4-5 *Learning-possibility that neuron model can recognize rotation in three-dimension* ·····486

Y. Sekiya, Q. Wang, T. Aoyama (Miyazaki University, Japan)
H. Tamura (Asahi Engineering, Co. Ltd., Japan)
Z. Tang (Toyama University, Japan)

WC4-6 *Autopoietic active selection by cognitive self-system : a unified theory for the origin and evolution of hierarchical neural-network-type cognitive biosystem-machines* ·····490

K. Ohnishi, H. Shutou (Niigata University, Japan)

MF: Poster Session (Room Karinn)

MF: *Improved adaptive fuzzy control and its application in molten carbonate fuel cell* ·····496

X. J. Sun, G. Y. Cao, X. J. Zhu
(Shanghai Jiaotong University, China)

MF: *A novel genetic algorithm and its application in control of molten carbonate fuel cell* ·····500

X. J. Sun, G. Y. Cao, X. J. Zhu
(Shanghai Jiaotong University, China)

MF: *A new method for designing robust neural network controller* ·····504

H. Chen, K. Hirasawa, J. Hu, J. Murata
(Kyushu University, Japan)

MF: *Modeling and evolutionary learning of modular neural networks* ·····508

Q. Zhao (The University of Aizu, Japan)

MF: *$H\infty$-control for fuzzy time-delay systems* ·····512

J. Yoneyama (Aoyama Gakuin University, Japan)

MF: *Function of general regularization term: case study on two-spiral classification problem* ·····516

W. Wan, K. Hirasawa, J. Murata, J. Hu
(Kyushu University, Japan)

MF: *Function approximation using LVQ and fuzzy set* ·····520

M. K. Shon, J. Murata, K. Hirasawa
(Kyushu University, Japan)

MF: *Robot path planning for visiting FA working points by obstacle avoiding using GA* ·····524

H. Yamamoto (Gifu University, Japan)
D. Moldovan (Wakayama University, Japan)

MF: *Self-organizing map neural network as a multiple model identifier for time-varying plants* ·····528

A. Fatehi, K. Abe (Tohoku University, Japan)

TF: Poster Session (Room Karinn)

TF: *An in vivo approach to molecular computation* ……532

K. Matsuno, M. Yamamoto, T. Shiba, A. Ohuchi
(Hokkaido University, Japan)

TF: *Music and meta-nature A short survey of alife music* ……536

R. Berry (ATR, Japan)

TF: *Learning motion coordination algorithm for distributed agents in cellular warehouse problem* ……539

K. Hama (Hakodate College of Technology, Japan)
S. Mikami, K. Suzuki (Future University-Hakodate, Japan)
Y. Kakazu (Hokkaido University, Japan)

TF: *Integrating top-down processing and bottom-up processing to interpret ambiguous figures* ……543

Y. Ohtsuka, N. Sugie (Meijo University, Japan)
H. Kudo, N.Ohnishi (Nagoya University, Japan)

TF: *Detecting eyes using circular hough transform and template matching* ……547

M. Rizon, T. Kawaguchi (Oita University, Japan)

TF: *A concideration on photovoltaic power generation systems* ……551

M. Sugisaka, K. Nakanishi, N. Mitsuo
(Oita University, Japan)

TF: *A robot location solution in a certain circumstance* ……555

Z. Li, J. Pei, H. Wu(Tsinghua University,China)

TF: *The running experiment of the wheel type mobile robot* ……558

M. Sugisaka, H. Aito (Oita University, Japan)

TF: *Reinforcement Leaning using a Gauss-Sigmoid Neural Network* ……562

S. Maehara, M. Sugisaka, K. Shibata (Oita University, Japan)

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Path Planning of a Mobile Robot by Optimization and Reinforcement Learning

Harukazu IGARASHI

School of Engineering, Kinki University
1 Takaya-Umenobe, Higashi-Hiroshima, Hiroshima 739-2116 Japan
E-mail: igarashi@hiro.kindai.ac.jp

In AROB5th, we proposed a solution to path planning of a mobile robot. In our approach, we formulated the problem as a discrete optimization problem at each time step. To solve the optimization problem, we used an objective function consisting of a goal term, a smoothness term and a collision term. While the results of our simulation showed the effectiveness of our approach, the values of weights in the objective function were not given by any theoretical method. This paper presents a theoretical method using reinforcement learning for adjusting the weight parameters. We applied Williams's learning algorithm, episodic REINFORCE, to derive a learning rule for the weight parameters. We verified the learning rule by some experiments.

Key Words: Path planning, Mobile robot, Optimization problem, Reinforcement learning

1 Introduction

There have been many works on the navigation [1] and motion planning [2] of mobile robots. For motion planning, property of a problem depends on the constraints imposed on the motion of a robot. This includes bounds on the velocity or acceleration of a robot as well as constraints on the curvature of paths. Moreover, different tasks require different criteria for estimating paths or trajectories. For example, users of an automatic motion planner may require the shortest path or the safest path depending on the task the user wants to make the robot do.

In a previous paper, we proposed a solution to navigation and motion planning of a mobile robot[3]. Our approach for navigation can be regarded as an approximation of Markov localization[4]. For motion planning, we defined and minimized an objective function that includes a goal term, a smoothness term and a collision term.

While the results of our simulation showed the effectiveness of our approach, the values of weights in the objective function were given only by a heuristic method. This paper presents a theoretical method using reinforcement learning for adjusting the weight parameters used in the action decision of a robot. In our reinforcement learning, a value function is defined by the expectation of a reward given to a robot's trajectory. Trajectories are generated stochastically because we used a probabilistic policy with a Boltzmann distribution function for determining the actions of the robot. This probabilistic policy optimizes the local objective function stochastically to search for the globally optimal trajectory. However, the stochastic process is not a Markov decision process because the objective function includes an action at the preceding time in the smoothness term. The usual Q-learning method cannot be applied to such a non-Markov decision process. Thus, we applied Williams's learning algorithm, episodic REINFORCE[5], to derive a learning rule for the weight parameters. The learning rule obtained here maximizes the value function stochastically. That was verified by our experiments.

2 Objective Function for Path Planning

We assume that a series of actions decided at each time step would derive a desirable path or trajectory of a robot. We define the following objective function E_v of a velocity v_t at time t,

$$E_v\left(v_t; r_t, v_{t-1}, r_{goal}\right) = b_1 E_{goal} + b_2 E_{smth} + b_3 E_{clsn}. \quad (1)$$

The minimal solution of Eq. (1) gives an estimate for the robot's action at time t.

The first term in the right-hand side of Eq. (1), E_{goal}, represents an attractive force to the goal r_{goal}. It is defined as

$$E_{goal}(v_t; r_t, r_{goal}) \equiv \mathrm{sgn}\left[G(v_t)\right] \cdot G(v_t)^2, \quad (2)$$

where sgn(x) denotes the sign of x and $G(v_t)$ is defined by

$$G(v_t) \equiv \left\| r_{goal} - r'_{t+1}(v_t; r_t) \right\| - \left\| r_{goal} - r_t \right\|. \quad (3)$$

The position r'_{t+1} is the robot's position at time t+1 if the robot moves at the velocity v_t from the position r_t. The second term in Eq. (1), E_{smth}, is defined by

$$E_{smth}(v_t; v_{t-1}) \equiv \left\| v_t - v_{t-1} \right\|^2 \quad (4)$$

to minimize changes in a robot's velocity vector. The last term in Eq. (1), E_{clsn}, represents a repulsion force for avoiding collisions with obstacles and walls. We define the term as

$$E_{clsn}(v_t;r_t)=\begin{cases} D_{clsn} & \text{if } Dist(r'_{t+1})<0 \\ -Dist(r'_{t+1})^2 & \text{if } 0<Dist(r'_{t+1})<L \\ -L^2 & \text{if } Dist(r'_{t+1})>L \end{cases}, \quad (5)$$

where Dist(r) means the shortest distance from the robot's position r to obstacles and walls. The constant D_{clsn} represents a degree of penalty given when the robot collides with obstacles or walls. It is usually set to a large positive value. The size L means a range within which the repulsion force starts to work on a robot. The strength of the repulsion force is calculated by using the distance to obstacles or walls not from the present position r_t but from the predicted position r'_{t+1}. In our simulation in Section 5, we set D_{clsn}=100000 and L=15.

Thrun et al. proposed another optimization approach for motion planning [6]. In their approach, a non-Markov term, such as E_{smth} shown in Eq. (4), was not taken into account.

3 Search Space

We restricted and discretized a search space when minimizing the function $E_v(v_t)$. We only took into account velocity vectors whose lengths were smaller than a constant. Figure 1 shows the search space used in our simulation in Section 5.

4 Learning Weights of Terms

4.1 Value Function and Probabilistic Policy

Shapes of trajectories given by minimizing $E_v(v_t)$ depend on weights of terms, $\{b_j\}(j=1,2,3)$[3]. In order to control the weights properly, we applied a reinforcement learning, episodic REINFORCE[5], which was proposed by Williams in 1992, to motion planning.

We define a value function $V(\pi)$, which is an expectation of a reward given to a trajectory l_i, as

$$V(\pi)\equiv E[R(l_i)] \quad (6)$$

$$=\sum_l P(l_i)\cdot R(l_i), \quad (7)$$

where $P(l)$ is a probability that trajectory l is produced by a policy π and $R(l)$ is a reward function representing user satisfaction to a trajectory l. A trajectory l_i is a series of robot's positions at times $t(t=0,1,...,N_i)$.

We define the policy π using a Boltzmann distribution function as

$$\pi(v_t;r_t,r_{t-1},\{b_k\})\equiv\frac{e^{-E_v(v_t)/T}}{\sum_{v_t} e^{-E_v(v_t)/T}}, \quad (8)$$

where $E_v(v_t)$ is the ojective function shown in Eq. (1) and T is a parameter to control the randomness in choosing an action v_t at each time. If we set T=0, it represents a deteministic decision process where every v_t gives the minimal solution of E_v in Eq. (1) at every time step. On the contrary, a completely randomized selection of actions occurs when T=∞.

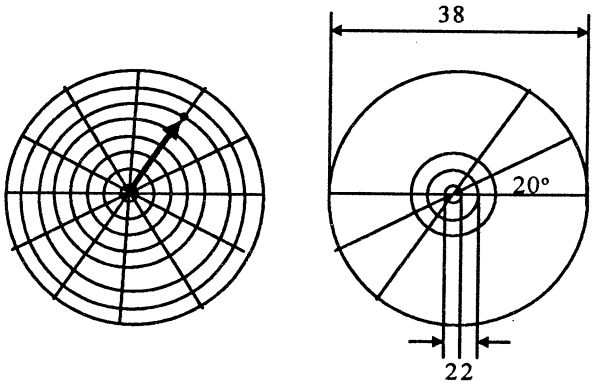


Fig. 1 Area searched for the optimal velocity of a robot

In Williams's episodic REINFORCE algorithm[5], he used a neural network model for a policy π. However, we used a Boltzmann distribution function that includes an objective function $E_v(v_t)$ as en energy function. The objective function can represent a heuristics for action decision and reduce the number of parameters.

4.2 Steepest Gradient Method

We use the steepest gradient method for maximizing the value funciton in Eq. (7). We have to caluculate the right-hand side of the following equation:

$$\frac{\partial}{\partial b_k}V(\pi)=\sum_l\left\{\frac{\partial}{\partial b_k}P(l_i)\right\}R(l_i). \quad (9)$$

The probability distribution $P(l_i)$ is expressed by a product of $P^\pi(r_t,r_{t+1})$'s as

$$P(l_i)=P^\pi(r_0,r_1)P^\pi(r_1,r_2)\cdots P^\pi(r_{N_i-1},r_{N_i}), \quad (10)$$

where $P^\pi(r_t,r_{t+1})$ is a probability that a robot moves to position r_{t+1} when it stands at r_t and takes a policy π. By differentiating the right-hand side of Eq. (10) with b_k, we obtained

$$\frac{\partial}{\partial b_k}P(l_i)=\sum_{t=0}^{N_i-1}\left[\prod_{n=0,n\neq t}^{N_i-1}P^\pi(r_n,r_{n+1})\cdot\frac{\partial}{\partial b_k}P^\pi(r_t,r_{t+1})\right]. \quad (11)$$

By substituting the following equation into the right-hand side of Eq. (11),

$$P^\pi(r_t,r_{t+1})=\sum_{v_t}P^{v_t}(r_t,r_{t+1})\cdot\pi(v_t), \quad (12)$$

we obtained that

$$\frac{\partial}{\partial b_k}P^\pi(r_t,r_{t+1})=\sum_{v_n}P^{v_t}(r_t,r_{t+1})\left\{\frac{\partial}{\partial b_k}\pi(v_t;r_t,r_{t-1},\{b_k\})\right\}$$

$$=\sum_{v_t}P^{v_t}(r_t,r_{t+1})\pi(v_t)\cdot\frac{\partial}{\partial b_k}\left[\ln\pi(v_t;r_t,r_{t-1},\{b_k\})\right]$$

$$=\sum_{v_t}P^{v_t}(r_t,r_{t+1})\pi(v_t)e_k(t), \quad (13)$$

where $e_k(t)$ is called the *characteristic eligibility* of b_k[5]. Substituting Eq. (13) into the right-hand side of

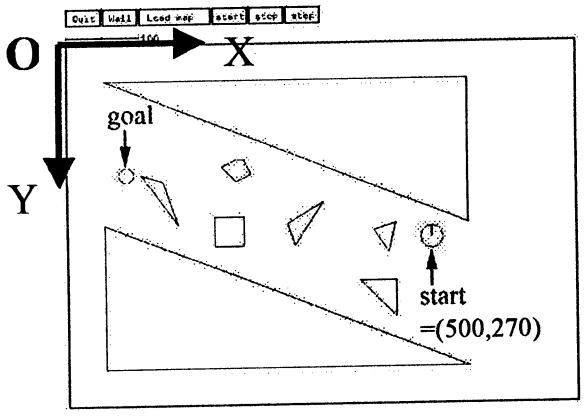


Fig. 2 Robot's world in our simulation

Eq. (11), we obtained from Eq. (9) that

$$\frac{\partial}{\partial b_k}V(\pi)=E\left[R(l_i)\cdot\sum_{t=0}^{N_l-1}e_k(t)\right] \tag{14}$$

This is the same result that Williams derived as his episodic REINFORCE algorithm [5].

However, we obtained a different updating rule from the REINFORCE algorithm because we used Eq. (8) instead of a neural network model for action decision. Using Eqs. (1) and (8), the characteristic eligibility of b_k is expressed as

$$e_k(t)\equiv\frac{\partial}{\partial b_k}\left[\ln\pi\left(v_t;r_t,r_{t-1},\{b_k\}\right)\right] \tag{15}$$

$$=-\frac{1}{T}\left[\frac{\partial E_v}{\partial b_k}-\left\langle\frac{\partial E_v}{\partial b_k}\right\rangle_{T,\{b_k\}}\right], \tag{16}$$

where operation <...> refers to the expectation weighted with a Boltzmann factor, i.e.,

$$\langle X\rangle_{T,\{b_k\}}\equiv\frac{\sum_{v_t}X\cdot e^{-E_v(v_t;r_t,r_{t-1},\{b_k\})/T}}{\sum_{v_t}e^{-E_v(v_t;r_t,r_{t-1},\{b_k\})/T}}. \tag{17}$$

By substituting Eq. (16) into $e_k(t)$ in Eq. (14) and using the steepest gradient method, we can derive the following learning rule of weights $\{b_k\}(k=1,2,3)$:

$$\Delta b_k=+\varepsilon\frac{\partial V(\pi)}{\partial b_k} \tag{18}$$

$$=-\frac{\varepsilon}{T}E\left[R(l_i)\cdot\sum_{t=0}^{N_l-1}\left\{\frac{\partial E_v}{\partial b_k}-\left\langle\frac{\partial E_v}{\partial b_k}\right\rangle_{T,\{b_k\}}\right\}\right]. \tag{19}$$

The constant ε is a learning rate factor to be set at a positive small number.

Moreover, we found that the parameters b_k's converge without the averaging operator E[...] in Eq. (19) by analogy with the error back-propergation algorithm in a multilayer network model[7]. We used the learning rule without the operator E[...] in the experiments in the next section.

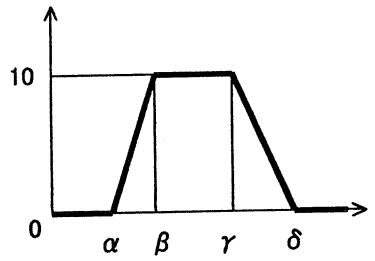


Fig. 3 General shape of achievement function

5 Simulation

5.1 Environment of a Robot

We consider a sample problem where a single robot moves to a goal from a starting position while avoiding static obstacles. Figure 2 shows the locations of obstacles, walls, the start point at (500,270), and the goal point at (80,180). In Fig. 2, a short circle having a radius of 8.0 and a short line drawn from the circle's center indicate the robot's position and front direction, respectively. Note that the robot has no body and that the short circle and line in Fig. 2 were used as a matter of convenience to show the robot's position and posture.

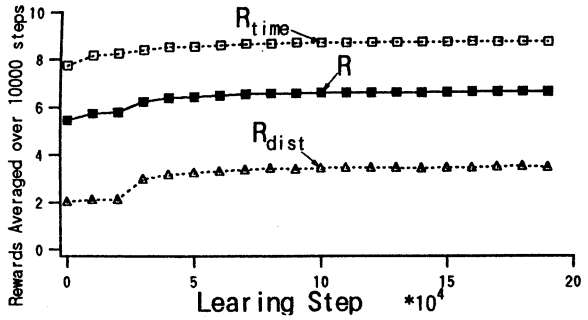
5.2 Reward Function

Let us assume that the objective of this sample problem is to find the trajectory that minimizes the robot's moving time from the start point to the goal point and keeps a safe distance from obstacles and walls. To find such a trajectory, we consider the following reward function $R(l)$ reflecting these two requirements:

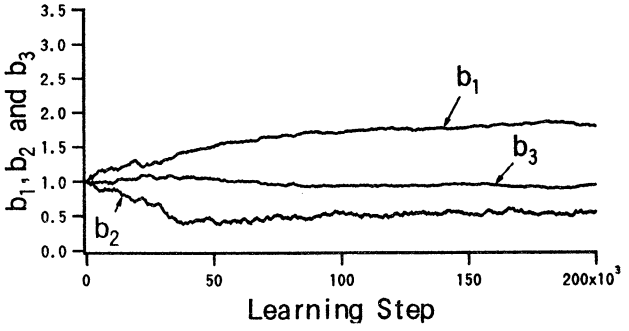
$$R(l)\equiv c_1R_{time}(l)+c_2R_{dist}(l), \tag{20}$$

where $R_{time}(l)$ represents the degree of user satisfaction to the moving time from the start to the goal in a trajectory l. The function $R_{dist}(l)$ represents the degree of user satisfaction to the shortest distance from the robot to obstacles and walls if a robot moves along a trajectory l. We call these functions *achievement functions* because they represent the degrees of achievement of user requirements on a trajectory. We give a large reward if the trajectory greatly satisfies the user of this planning system. We used the trapezoid-like functions shown in Fig. 3 as achievement functions. They take a value between 0 and 10 and are characterized by the parameters, α, β, γ and δ. The total reward R is obtained by averaging the values of the two achievement functions with weights c_1 and c_2.

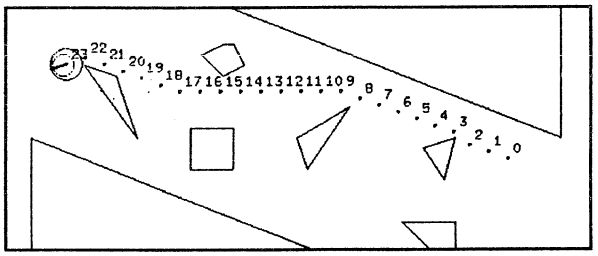
In our experiment, we set $\alpha=\beta=0$, $\gamma=20$,$\delta=50$ for $R_{time}(l)$ and $\alpha=0$, $\beta=15$, $\gamma=\delta=2000$ for $R_{dist}(l)$. This means that trajectories whose moving times are shorter than 20 and trajectories that force a robot to keep a distance longer than 15 are most desirable.



(a)

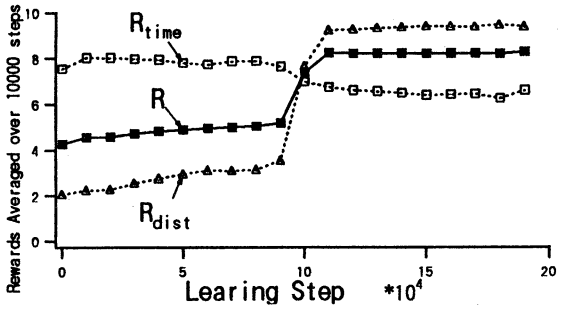


(b)

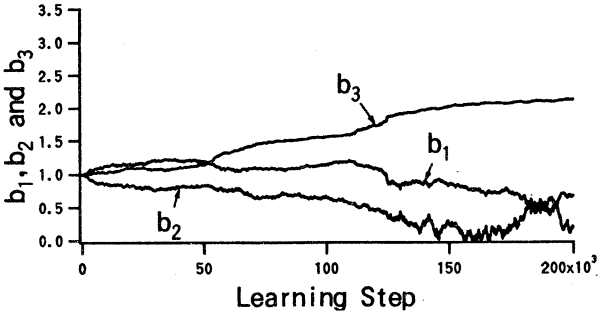


(c)

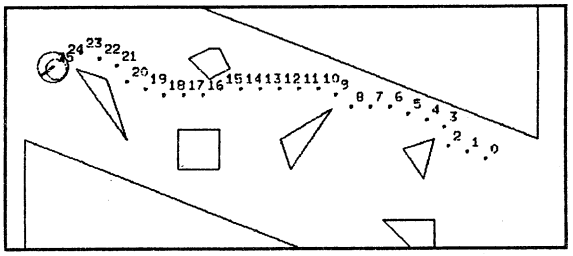
Fig.4 Results of experiment I, when c_1=0.6 and c_2=0.4: (a) R, R_{time} and R_{dist} averaged over ten thousand learning steps (b) b_1, b_2 and b_3 (c) The trajectory obtained deterministically after learning $\{b_j\}$ (j=1,2,3) two hundred thousand times



(a)



(b)



(c)

Fig.5 Results of experiment II, when c_1=0.4 and c_2=0.6: (a) R, R_{time} and R_{dist} averaged over ten thousand learning steps (b) b_1, b_2 and b_3 (c) The trajectory obtained deterministically after learning $\{b_j\}$ (j=1,2,3) two hundred thousand times

5.3 Experimental Conditions

We set initial values of b_k's as b_1=b_2=b_3=1.0. Parameter T is set at 5.0 and not decreased during the search. This means that we did not carry out any annealing procedure. A learning rate factor ε in Eq. (19) is set at 0.00001. The weights c_1 and c_2 in Eq. (20) are set as c_1=0.6, c_2=0.4 in experiment I and c_1=0.4, c_2=0.6 in experiment II.

Under these experimental conditions, we generated a trajectory using the policy π in Eq. (8), evaluated the reward to the trajectory by Eq. (20) and updated weights $\{b_k\}$(k=1,2,3) without carrying out the expectation operator E[...] in Eq. (19). We repeated this learning cycle two hundred thousand times in our experiment. It took about 200 minutes to complete one experiment using a work station, SUN Ultra Sparc 30 (CPU: Ultra Sparc-II, 248 MHz).

5.4 Experimental Results

Figure 4 shows the results of experiment I. In Fig.4(a), changes in R, R_{time} and R_{dist} are shown. The values are averaged over each period of ten thousand learning steps. Changes in parameters $\{b_k\}$(k=1,2,3) are shown in Fig. 4(b). The trajectory obtained deterministically with the weight parameters that had been learned two hundred thousand times is shown in Fig.4(c). Figure 5 shows the results of experiment II.

Figure 6 shows the trajectory obtained using unlearned values. When obtaining the trajectories shown in Fig.4(c), Fig.5(c) and Fig.6, the parameter T was set to a very small positive. They were equal to trajectories given by a deterministic theory that minimizes $E_v(v_t)$ at each time step. The probabilistic policy π with a finite value of T makes the trajectory given by a deterministic theory fluctuate. The

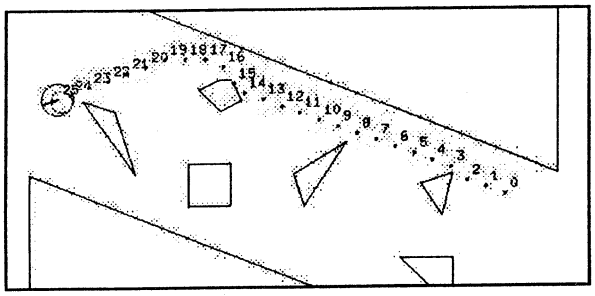


Fig. 6 The trajectory obtained deterministically with weights that had not been learned

fluctuation contributes to searching for the optimal trajectory that maximizes the value function $V(\pi)$ in Eq. (6).

5.5 Discussion

In experiment I, the weight b_1 of the goal term increased gradually. That reduced the travelling time from 25 steps to 23 steps. In experiment II, the problem that the robot ran into an obstacle at time 15 shown in Fig.6 was solved by learning without delaying the robot's arrival time. This improvement comes from the improvement in the second term R_{dist} in Eq. (20). The function R_{dist} forces a robot to keep a certain distance from obstacles and walls for safety. Figure 5(b) shows that this improvement was caused by the increse of the value of b_3, which is the weight of the collision term E_{clsn} in $E_v(v_t)$.

Moreover, the trajectory was not only improved locally. A global change in trajectory was achieved. That is because the weights $\{b_k\}$ in Eq.(1) do not depend on the robot's position. Thus, if we would like to change trajectories locally to search for the optimal trajectory, we had better consider term weights $\{b_k(r)\}(k=1,2,3)$ that depend on the robot's position r. Our learning rule can be applied to $b_k(r)$ in the same way as applied to b_k.

6 Terms to Escape from Local Minima

The objective function defined in Eq.(1) sometimes misleads a robot to a local minimum if the arrangement of obstacles is complicated as a maze. We can add a term to $E_v(v_t)$ so as to escape from local minima. We will show two examples.

The first example is E_{esc1} that is defined by,

$$E_{esc1}(v_t;r_t)=\vec{b}_4(r_t)\cdot v_t, \qquad (21)$$

where the local vector field $b_4(r)$ shows a desirable direction at position r. The second example is a term that includes attractive forces to sub-goals. For example, it is defined by

$$E_{esc2}(v_t;r_t)=\sum_{r_{sgoal}} b_5(r_{sgoal})E_{goal}(v_t;r_t,r_{sgoal}), \qquad (22)$$

where r_{sgoal} are sub-goals that are located at some intersections of channels to help a robot to escape from local minima.

7 Conclusions and Future Work

This paper presented a theoretical method using reinforcement learning for adjusting the weight parameters in the objective function that was used in motion planning[3]. This optimized the local objective function at each time step stochastically to enable search for the globally optimal trajectory. We applied Williams's learning algorithm, episodic REINFORCE, to derive a learning rule for the weight parameters in the objective function. Moreover, a stochastic hill-climbing method can be applied to our learning rule and our method can deal with non-Markov decision processes while the conventional Q-learning method can be applied only to Markov decision processes. The learning rule proposed here was verified by some experiments.

We plan to apply our method to motion planning problems of multi-robot systems. In a multi-robot system, interactions between robots can be expressed as terms in the objective function $E_v(v)$. If a user wants to move two robots keeping them close to or avoiding each other, it is sufficient to introduce an attractive or repulsion force between robots into the objective function $E_v(v)$. This shows the flexibility that our method can be applied to many scheduling problems in the wide-range field of engineering.

References

[1] Kortenkamp D, Bonasso RP, Murphy R(1998), Artificial Intelligence and Mobile Robots. AAAI Press/The MIT Press , 1998.

[2] Hwang YK, Ahuja N(1992), Gross Motion Planning—A Survey. ACM Computing Surveys 24(3):219-291

[3] Igarashi H, Ioi K(2000), Path Planning and Navigation of a Mobile Robot as Discrete Optimization Problems. In:Sugisaka M, Tanaka H(eds), Proceedings of the Fifth International Symposium on Artificial Life and Robotics (AR OB5th), Oita, Japan, Jan26-28, 2000, pp. 297-300

[4] Burgard W, Fox D, Thrun S(1997), Active Mobile Robot Localization. Proceedings of 15th Joint Conference on Artificial Intelligence (IJCAI97), Nagoya, Japan, August 23–29, 1997, Vol.2, pp. 1346-1352

[5] Williams RJ(1992), Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning. Machine Learning 8:229- 256

[6] Thrun S et al, Map Learning and High-Speed Navigation in RHINO, in [1], pp. 21-52.

[7] Amari S(1967), A Theory of Adaptive Pattern Classifiers. IEEE Trans. EC-16:279-307

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Integrated Path Planning and Steering Control with Multisensor Fusion for Intelligent Mobile Robots

○Hiroyuki Takai*, Gen'ichi Yasuda**, and Keihachiro Tachibana*

*Department of Information Machines and Interfaces
Faculty of Information Sciences
Hiroshima City University
3-4-1 Ozuka-higashi, Asaminami-ku
Hiroshima 731-3194, Japan
E-mail: takai@im.hiroshima-cu.ac.jp

**Department of Mechanical Engineering
Faculty of Engineering
Nagasaki Institute of Applied Science
536 Aba-machi, Nagasaki 851-0193, Japan
E-mail: yasuda@csce.nias.ac.jp

Abstract: This paper presents a control scheme for intelligent motion control of mobile robots under unstructured environments. Based on environmental information, which is acquired using external sensors or given by the upper level, the control scheme performs autonomous navigation composed of path planning and tracking control in real time. First, using a simple genetic algorithm, the path planning module calculates an obstacle-free path as a sequence of control vectors of orientation, considering kinematic constraints in steering control of wheeled mobile robots. Then, the tracking control module calculates the references for motion control of the mobile robot using a neural network. Simulation experiments of path planning under unstructured environments with several obstacles are illustrated. An experimental procedure for teaching the neural network is introduced, and basic characteristics of the internal and external sensors during straight-line and circular movements on the floor with black-striped marks are measured using our experimental small robot to show the effectiveness of the proposed control scheme.

Keywords: Mobile robots, path planning, genetic algorithms, path tracking, neural networks, gyro sensors.

1. INTRODUCTION

One of the most fundamental problems in the design and development of intelligent mobile robots is the navigation problem under dynamically changing environments. Generally, the navigation task is composed of path planning, path-following trajectory generation, and tracking control. When the robot environment is dynamic, like as robotic soccer playground, the path planning and trajectory generation algorithm should be simple and efficient enough to be used in real time by onboard computer systems [1], [2].

In this paper, the path planning and the tracking control are decoupled using a two-level hierarchical strategy, but integrated considering kinematic constraints in steering control of nonholonomic two-wheels vehicles. At the beginning of planning cycle, the control system finds objects within limited sensing range using a local vision system, and decides the goal position and orientation. If there are objects between the robot and its goal, the path planning module calculates an obstacle-free path from the current configuration (position and orientation) to the goal configuration as a set of via-points, or control vectors of orientation using an efficient genetic searching algorithm. Then, the tracking control module performs the steering control for the path tracking. Conventionally, path tracking is executed only based on position and velocity errors. Conversely, references for steering control are produced by learning using a neural network based on multisensor fusion of internal and external sensors, and they are recalled using the internal sensor data when external sensors can not be used. Preliminary experiments were conducted, where gyro sensors and infrared photoreflectors are used as internal and external sensors, respectively.

2. PATH PLANNING USING GENETIC ALGORITHM

A genetic searching algorithm for the path planning with obstacle avoidance was developed [3]. After finding the objects, the control system decides the goal position and orientation of the mobile robot. Then, the path planning module is activated to generate via-points for a short and safe path to the goal with stationary obstacle avoidance. In the genetic algorithm, a path is represented as a set of vectors of orientation with equal length. The mobile robot is assumed to move at constant linear velocity. The gene means the change of orientation of the path; the resultant path is composed of polygonal lines, where the number of via-points, or genes, is fixed. To reduce the length of code of the gene, the change of orientation is restricted to 5 values from 45 deg to -45 deg, considering kinematic constraints in steering control. The genes and an example of chromosome are shown in Fig. 1.

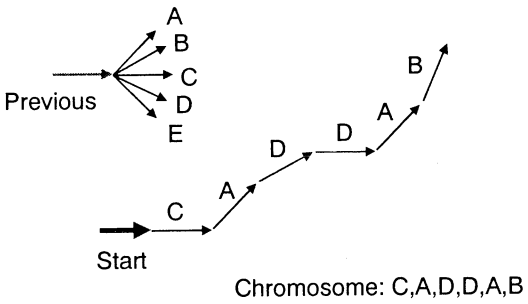


Fig. 1. Genes and example of chromosome.

The fitness value of a chromosome, or planned path, is defined using the distance parameters from the path to the goal and obstacles as follows:

$$fittness = \alpha \frac{L}{\sum_{i=1}^{N} l_i + l_{end}} + \beta \frac{\sum_{i=1}^{N} d_i}{NL} + \gamma \cdot f \frac{x}{l_{end}}$$

where

L : distance between start position and goal position

l_i : distance between i th control point and goal position

l_{end} : distance between N th via-point and goal position

d_i : minimum distance between i th via-point and obstacles

N : number of genes

x : projection from N th via-point to goal orientation

f : if orientation at N th via-point coincides with goal orientation, $+1$; otherwise -1

α, β, γ : weighting factors

The fitness value is a weighted sum of the following three terms: 1) the rate of distance of the planned path to that of the straight line both to the goal position, 2) the sum of distance between via-points and the nearest obstacles, and 3) the deviation from the goal orientation (Fig. 2).

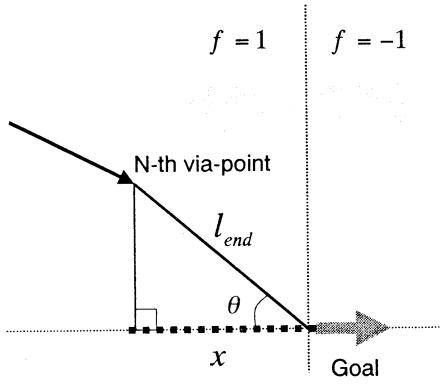


Fig. 2. Deviation from goal orientation.

Conventional GA operators such as one-point crossing over, point mutation, and combined roulette and elite selection regime, were applied. For real-time consideration, the total generation number is fixed to a value required for the convergence of the fitness value. Path planning is re-initiated before the robot reaches the final control points to perform smooth movements without stopping. An example of planned path is illustrated in Fig. 3. The genetic algorithm parameters are listed in Table 1. The mean processing time for path planning was 0.48 sec using SunUltra60 (300 MHz UltraSparc II CPU, with 256 Mbytes of memory).

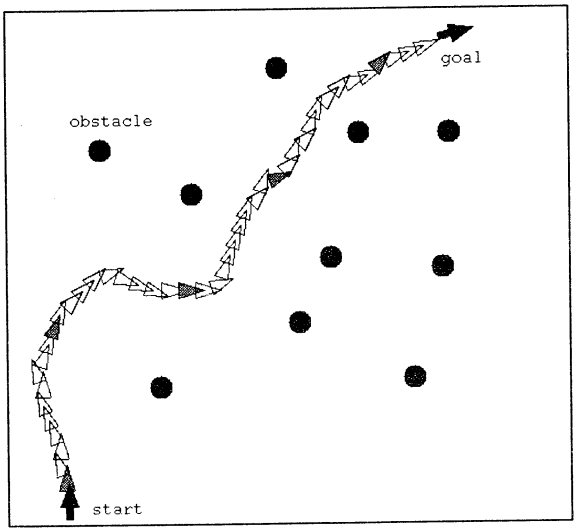


Fig. 3. Example of planned path.
($\alpha = 11, \beta = 10, \gamma = 5$)

Table 1 Genetic algorithm parameters

Population size	80
Generation number	100
Gene number	6
Crossover probability	0.8
Mutation probability	0.05

The relative amounts of the three terms change according to different environmental conditions. For satisfactory convergence of the algorithm, the weighting factor β is set to a relatively large value if the distance between the start position and the goal position is long, because the first and third terms are relatively large. On the other hand, the weighting factors α and γ are set to relatively large values if the distance is short. For moving obstacles, naturally, shorter planning cycle and larger sensing range are required, and the linear velocity should be also changed. Furthermore, the weighting factor β should be set to a relatively large value for safe path consideration. The weighting factors should be adjusted according to different environmental conditions using some neural network.

3. TRACKING CONTROL SCHEME

From the path planning module, current references, or distance and orientation is given to the tracking control

module. Then, it calculates the desired control variables for the motors of the mobile robot using a neural network at each control cycle. The overall block diagram of the proposed tracking control scheme is shown in Fig. 4.

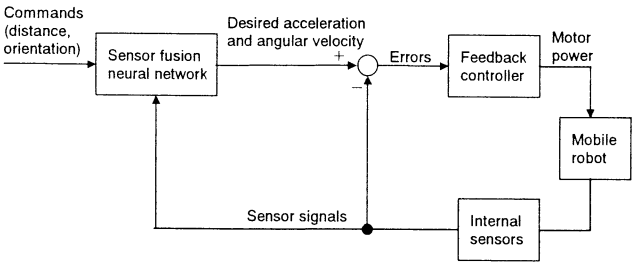


Fig. 4. Block diagram of tracking control scheme.

The network inputs are the reference distance and orientation, and the output signals of internal sensors. As internal sensors, a piezoelectric vibratory gyroscope (Tokin CG-16D) and two orthogonally mounted variable capacitance type linear accelerometers (Analog Devices ADXL202) are used, because the odometry using rotary encoders bring about large errors because of wheel slippage. The network outputs are the desired output values of the internal sensors, and their measured values are fed back into the conventional feedback controller to calculate manipulated variables, or PWM duty cycle, of motor power applied to the DC motors of a two-wheels mobile robot. The tracking controller does not need to calculate the position and orientation of the mobile robot, because integral of inertial sensor data induces rather large errors.

In the training phase of the neural network, multiple internal and external sensors are fused. External sensors measure the displacement and linear and angular velocities of the mobile robot, and at the same time internal sensory information is memorized as the reference for robot motion control such as steering control. The reference values are used in path tracking without external sensors when commands are given in terms of distance and orientation, or curvature of the path, from the upper level.

An experimental procedure for training the neural network is designed as follows. The linear and angular velocities are measured using an infrared sensor system by moving on the floor with black-striped marks. The measured velocities are fused with the current outputs of the internal sensors, and the network is trained using the back propagation method to produce desired outputs. The proposed control software was implemented on a microcontroller-based control system (Hitachi 16MHz H8-3048 CPU) under a real-time OS, with the interface circuits for the internal and external sensors as shown in Fig. 5.

4. PRELIMINARY EXPERIMENTS

Our experimental mobile robot is two independently driving wheels type. A DC motor is geared to each wheel, and a PWM drive is used to control power applied to the motors. The wheel diameter is 65 mm, and the wheel separation distance is 160 mm. Fig. 6 sketches the experimental set up, and Fig. 7 shows the mobile robot moving on the floor with black-striped marks.

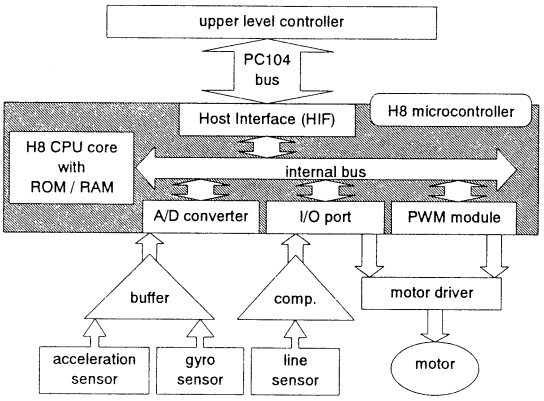


Fig. 5. Hardware configuration of tracking control system.

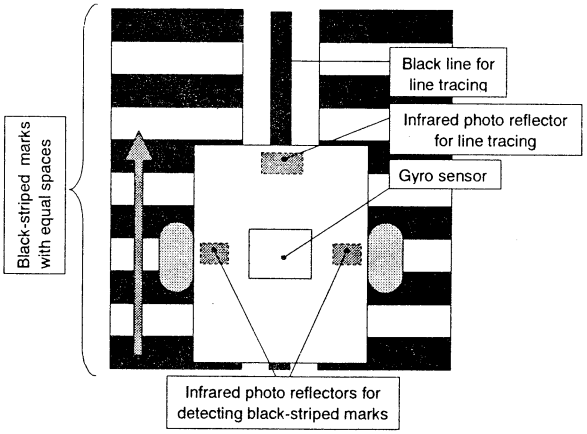
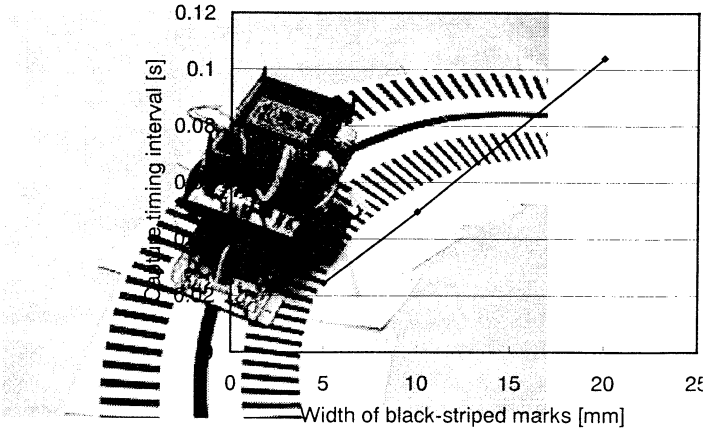


Fig. 6. Sketch of experimental set up.

Fig. 7. Circular movement on the floor with black-striped marks.



The mobile robot detects black-striped marks with equal spaces of 10 mm during line tracing using dedicated

OMRON EE-SY121 infrared photo reflectors. The velocities are measured by the input capture module of the microcontroller. Fig. 8 shows the relationship between duty cycle and measured linear velocity. The angular velocity can be measured using the difference of capture timing intervals from two photo reflectors. Table 2 shows the measurements during the straight-line and curved-line movements, where the linear velocity is measured using the photo reflectors. The theoretical values of gyro sensor measurement, which are calculated from the measured linear velocity, the radius of curvature of the circle and the electric specifications of the sensor, are in good agreement with the measured ones, although the accuracy of the gyro sensor was ± 20 %.

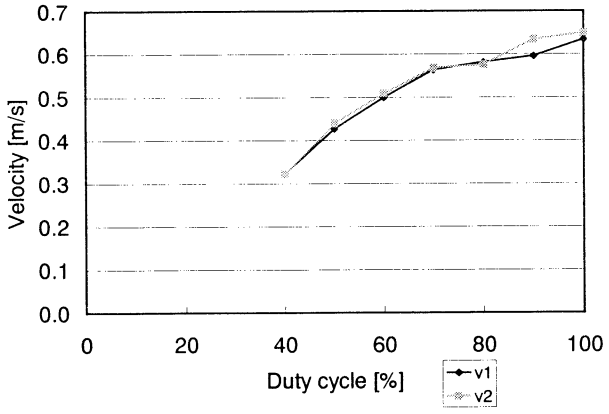


Fig. 8. Relationship between duty cycle and measured linear velocity: v1= human eye measurement, v2= measurement by the external sensors.

Table 2 Measurements during moving on black-striped marks

	Linear velocity [m/s]	Gyro sensor output [V]
Straight-line	0.379	2.55
Curved-line (clockwise)	0.376	2.39
Curved-line (counterclockwise)	0.383	2.73

(PWM period = 10 ms; PWM duty cycle = 50 %; circle radius = 0.5 m; space between marks = 10 mm)

Fig. 9 shows the values of capture timing interval measured by the input capture module during moving at constant speed on black-striped marks with different widths. From the experimental results, it is confirmed that the infrared sensor system can measure widths of black-striped marks accurately, and the internal sensor system can be used to generate the references for motion control.

Fig. 9. Capture timing interval during moving on black-striped marks with different widths.

5. CONCLUSIONS

An integrated path planning and steering control scheme was proposed for autonomous navigation of intelligent mobile robots. The control scheme is simple and efficient enough to be used in real time. Straight-line and circular movements on black-striped marks are used for teaching the neural network, where the internal sensors are fused with the external sensors. The experimental results show the effectiveness of the proposed measurement system using the infrared photo reflectors and the gyro sensor. The whole control scheme is now being implemented on the microcontroller-based architecture.

ACKNOWLEDGMENT

This research is supported, in part, by the Grant-in-Aid of Scientific Research from the Japan Society for the Promotion of Science (No. 11680408), and, in part, by Hiroshima City University Grant for Special Academic Research (Encouragement for Researchers No.9858, 9958, 0059).

REFERENCES

[1] Han WG, Beak SM, Kue TY (1996), Path planning of visual-servoed multi-mobile robots using the genetic algorithms. In: Proceedings of Micro-robot World Cup Soccer Tournament, Taejon, Korea, Nov 9-12, 1996, pp.57-63

[2] Yasuda G, Takai H (1999), A microcontroller-based architecture for locally intelligent robot agents (in Japanese). In: Proceedings of the 4th JSAI Sig-AI-Challenge, Tokyo, Japan, March 17, 1999, pp.11-15

[3] Shimazaki T, Takai H, Tachibana K (1999), Path planning for mobile robots with its directions using GA (in Japanese). In: Proceedings of the 5th JSAI Sig-AI-Challenge, Nagoya, Japan, May 1, 1999, pp.13-15

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Environment Mapping for Khepera Robot: A New Method by Fusion of Pseudo Information Measures

Mohammad Reza Asharif *
Professor
Department of Inf. Engineering
Faculty of Engineering
University of the Ryukyus
JAPAN
asharif@ie.u-ryukyu.ac.jp

Behzad Moshiri†
Associate Professor
ECE Department
Faculty of Engineering
University of Tehran
IRAN
moshiri@karun.ipm.ac.ir

Reza HoseinNezhad ‡
Ph.D. Student
ECE Department
Faculty of Engineering
University of Tehran
IRAN
rhn75@dsp.ie.u-ryukyu.ac.jp

Abstract

A new mathematical concept, named **Pseudo Information Measure** is introduced and some new formulations for sensor data fusion in mobile robotic map building are obtained. In some map building experiments the 8 proximity values of the infra red sensors of a Khepera robot are fed into the inputs of a neural network. The output of the network, is an occupancy probability value of a cell in the map which is fused with currently available probability value of the associated cell in a global map, to improve the global map of the environment. The fusion is done, by using Bayesian and our new fusion formulas. Some path planning experiments using a modified version of A* algorithm have been done. The resulting paths show that the maps, generated by our new fusion formulation can be more informative and lead to shorter or safer paths in an unknown environment for the robot to navigate.

Key words: Khepera, Environment Mapping, Perception, Sensor Data Fusion, Bayesian Rule of Combination

1 Introduction

Since mobile robots are typically equipped with several sensors of different modalities, they are suitable benchmarks to test the revenue of any sensor fusion method. For an autonomous mobile robot to operate in unknown environments, two main capabilities are required: the ability to reach a target while avoiding the obstacles it might find on the way, and the ability to build environment models from sensor observations. The sensory information, after some neural preprocessing, are some occupancy probability values. They are noisy, redundant and complement. These values can be integrated, by using Bayesian fusion technique ([1]), to give more certain and less noisy values for the cells in the map. In this research work we have improved the Bayesian formula by introducing a new concept and obtaining new formulas for effective fusion of sensory data.

1.1 Occupancy Grids and Bayesian Fusion

In occupancy grids framework, the environment area is divided into many small square cells. For each cell of the map two states are possible, either it is *occupied* or it is *empty*. The map is generated by measuring probability values for each cell in it i.e. by judging about the state of the cell. In Bayesian approach, the range finder sensor is modeled by some conditional distribution $P[r|z]$ where z & r are the actual and the measured distances, respectively ([1]). If multiple measurements are available (obtained by several sensors or by a single sensor but in several time instances), then by Bayesian rule in the probability theory, the map is improved. Practically it can be assumed that the sensors are operating independently. In such a case, the fusion formula is simplified as below:

$$P = \frac{P_1 \times P_2}{P_1 \times P_2 + (1 - P_1) \times (1 - P_2)} \qquad (1)$$

*Also assistant professor in department of electrical and computer engineering, faculty of engineering, university of Tehran

†Also researcher in School of Intelligent Systems (SIS), IPM, Tehran, IRAN

‡Also research student in department of information engineering, faculty of engineering, university of the Ryukyus

where

$P_1 = P[s(C_i) = Occ.|\{r_t\}]$
$P_2 = P[s(C_i) = Occ.|r_{t+1}]$
$\{r_t\}$ = all of the measurements up to time t
r_{t+1} = the measurement at time $t+1$

2 A New and Useful Concept

We tried to find a quantity, that is increased while the fusion takes place by (1). It can be interpreted as some quantitative description of the information existing in the proposition C_i is occupied. If a function, INFO(P) is defined as below:

$$\text{INFO}(P) = \log(P) - \log(1-P) \tag{2}$$

then, combination of (2) and (1) simply results the following desired property for INFO(P):

$$\text{INFO}(P) = \text{INFO}(P_1) + \text{INFO}(P_2) \tag{3}$$

Figure 1 shows the mathematical shape of the function INFO(P). As P goes farther away from $\frac{1}{2}$ the value of |INFO(P)| increases because the proposition becomes more informative. But in order to distinguish between the propositions that are near to be `false` and the ones that are near to be `correct`, the function INFO(P) is symmetrically negative in the former case and positive in the latter. We tried to extend this concept to a more general case. Actually there are some properties that if every other function satisfies them, then it can be accepted as a quantitative

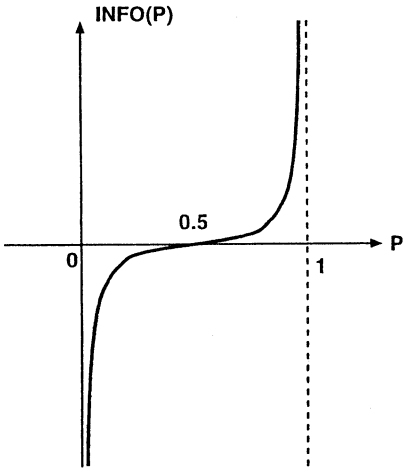


Figure 1: General mathematical shape of a pseudo-information function

measurement function for the information existing in a proposition. We nominate such functions as *pseudo information measure functions* and will use the symbol PINFO(P) for them. The desired properties are as below:

1- It is defined on $[0,1]$

2- It is symmetric around $\frac{1}{2}$ and zero at $\frac{1}{2}$. So, it can be defined as

$$\text{PINFO}(P) = J(1-P) - J(P) \tag{4}$$

Clearly in the case of (2) the function $J(P)$ is $-\log(P)$.

3- It must satisfy the following limits:

$$\begin{aligned} &\mathcal{L}im_{P\to 0+}\text{PINFO}(P) = \mp\infty \\ &\mathcal{L}im_{P\to 1-}\text{PINFO}(P) = \pm\infty \end{aligned} \tag{5}$$

and it is sufficient for the function $J(P)$ to satisfy the following limits:

$$\begin{aligned} &\mathcal{L}im_{P\to 0+}J(P) = \pm\infty \\ &\mathcal{L}im_{P\to 1-}|J(P)| < \infty \end{aligned} \tag{6}$$

4- It must be an increasing/decreasing (vice versa $J(P)$ must be a decreasing/increasing) function.

2.1 Data Fusion Using Pseudo Information Measure

Defining a fusion formula is straightforward after the definition of a pseudo information function. Two sources of information that have been processed and then expressed as two probability values for some proposition, are fused in such a way that the pseudo information measure for the resulting probability is the algebraic sum of the values associated with the two probabilities. It is the same as (3) rewritten as:

$$\text{PINFO}(P) = \text{PINFO}(P_1) + \text{PINFO}(P_2) \tag{7}$$

Hereby, there are six possible definitions for pseudo information measure function:

$$\begin{aligned} &J_1(P) = -\log(P) \quad J_2(P) = \tfrac{1}{P} \quad J_3(P) = \tfrac{1}{P^{1.1}} \\ &J_4(P) = \tfrac{1}{P^2} \quad J_5(P) = \tfrac{1}{e^{P-1}} \quad J_6(P) = \log(\tfrac{1+P^2}{P}) \end{aligned}$$

Figure 2 gives a comparative representation for the mathematical shapes of the six pseudo information measure functions created by the functions $J_1(.) \sim J_6(.)$. It compares the behaviour of the pseudo information values while $|P - \frac{1}{2}|$ increases. $J_1(.)$ is the function which leads to Bayesian fusion formula (1). It is apparent that in $J_2(.)$ and $J_3(.)$ and $J_4(.)$, we have tried to use fractional functions instead of the

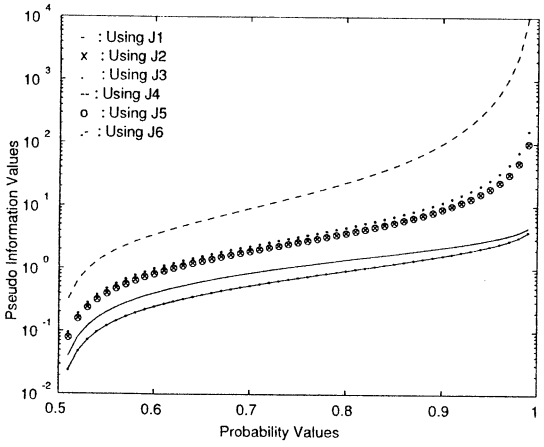


Figure 2: Comparative mathematical shapes of the 6 pseudo-information functions created by 6 functions $J_1(.)$-$J_6(.)$.

logarithmic one in $J_1(.)$, and in $J_5(.)$ and $J_6(.)$, the same trial happened for applying exponential and another logarithmic functions. By applying (4) and (7), one can easily obtain fusion formulas for every definition of pseudo information. In some cases, an explicit formula can be directly calculated. For example, in the case of applying $J_2(.)$ to calculate the pseudo information measure values, the following fusion explicit formula is attained:

$$P = \begin{cases} \frac{x-2+\sqrt{x^2+4}}{2x} & \text{if} \quad x \neq 0 \\ \frac{1}{2} & \text{if} \quad x = 0 \end{cases} \tag{8}$$

where $x = \frac{1}{1-P_1} - \frac{1}{P_1} + \frac{1}{1-P_2} - \frac{1}{P_2}$, or in the case of applying the exponential function of $J_5(.)$ the following formula is resulted:

$$P = \log\left(\frac{x(e+1)+\sqrt{x^2(e-1)^2+4e}}{(2(x+1))}\right) \tag{9}$$

where $x = J_5(1-P_1) - J_5(P_1) + J_5(1-P_2) - J_5(P_2)$. But in many cases, there is no direct formula for doing fusion and an implicit algebraic equation must be solved. For example in the case of $J_6(.)$, the following equation must be solved:

$$\frac{2P^2-p^3}{1-P-P^2+P^3} = \frac{2P_1^2-p_1^3}{1-P_1-P_1^2+P_1^3} + \frac{2P_2^2-p_2^3}{1-P_2-P_2^2+P_2^3} \tag{10}$$

Finally, as figure 2 shows, the main difference between the $J(.)$ functions, is in their behaviour when $|P-\frac{1}{2}|$ increases. Actually, it can be proved that:

$$|J_6(P)| \leq |J_1(P)| \leq |J_2(P)| \cong |J_5(P)| \leq |J_3(P)| \ll |J_4(P)|$$

and it shows that the pseudo information measure, calculated by using $J_6(.)$, behaves more softly with

Figure 3: The exact (ideal) map of the navigation environment.

low and high probability values (associated with more informative propositions) and the inverse second order function $J_4(.)$, behaves roughly about informative propositions and has not been considered in the comparative implementation in this research work.

3 Map Building Experiments

Khepera miniature mobile robot, was used to do some map building experiments. In figure 3 the exploration environment is depicted. An MLP neural network was supervised for map building. The inputs of the network are 8 proximity values, provided by Khepera infrared sensors, and the local coordinates of a cell in the occupancy grids map around the robot. The output of the network is one probability value, judging about the state of the cell to be occupied or empty. This probability value is fused with the associated occupancy probability value of the same cell in a global map of the environment and finally the resulting probability is applied to improve the global map.

In other words, during exploration, Khepera makes a local map of occupancy probabilities of the cells around itself, in each location where its infrared sensors data are read. These local maps are integrated with global map and it improves gradually.

Four map building experiments were done, by using Bayesian fusion (1) and three of pseudo information fusion formulas (8) and (9) and (10). In figures 4 and 5 the four resulting maps of the environment are shown. It is observed that the maps resulted from pseudo information fusion formulas are more informative and more useful for path planning purposes. Actually their characteristics depends on the pseudo in-

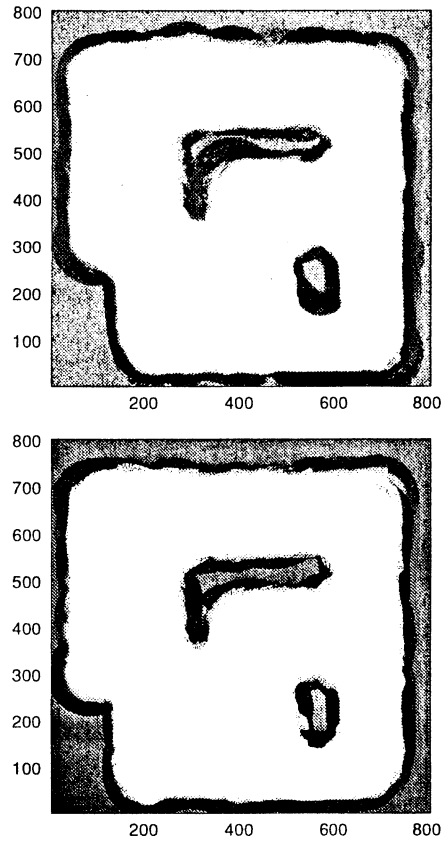


Figure 4: The global maps of the environment, created using Bayesian fusion (top) and pseudo information measure fusion by $J_2(P)$ (bottom).

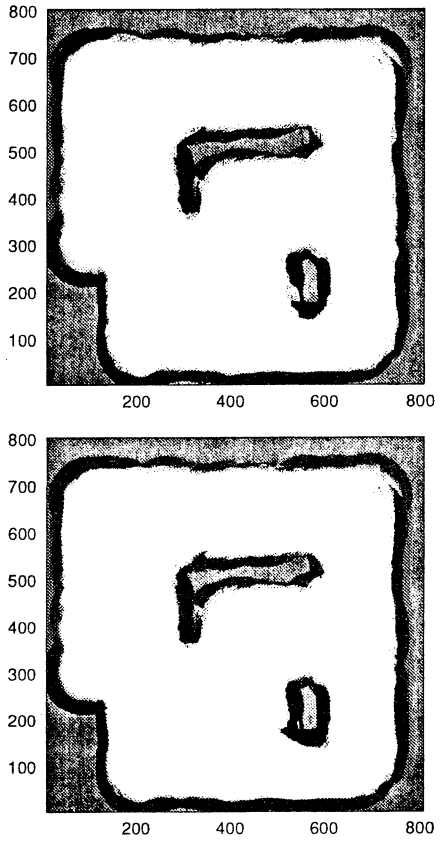


Figure 5: The global maps of the environment, created using pseudo information measure fusion by $J_5(P)$ (top) and by $J_6(P)$ (bottom).

formation measure function, selected for fusion. For example, compared to Bayesian map, the maps generated by $J_2(P)$ and $J_5(P)$ are more clear but less safe (i.e. with less area marked to be occupied) and the map generated by $J_6(P)$ seems to be not as clear as the others but more safe. Some path planning experiments, using an improved version of A* algorithm, were done using the generated maps. They show that the maps generated by $J_2(P)$ and $J_5(P)$, lead to shorter paths (compared to the case of using Bayesian map for path planning), and the map generated by $J_6(P)$, leads to safer paths.

4 Conclusion

In this research work, a new method for sensor fusion in map building process for mobile robots is introduced. Because of the large variety of possible definitions, many mathematical formulations are possible. Map building and path planning implementation results show that by choosing the the pseudo information function, we can control the existing trade off between the traveling time (path length) and safety (path distance to obstacles).

Acknowledgements

This work was partially supported by SIS (School of Intelligent Systems) at IPM, Tehran, IRAN under grant number 0177-115. The grant provided by Japanese MONBUSHO scholarship is also kindly appreciated

References

[1] Elfes A. (1989), Using Occupancy Grids for Mobile Robot Perception and Navigation. Computer 22(6), pp. 249-265

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

The Controlling of The Welfare Robot Prototype

Masanori Sugisaka Takuya Adachi
Department of Electrical and Electronic Engineering
Oita University
700 Ohaza Dannoharu, Oita, 870-1192
E-mail: {msugi, adachi}@cc.oita-u.ac.jp

1. Abstract

Today, in Japan, the aging society that will happen to the future becomes an important problem. To do welfare the old persons who increasing every year that needs. But, to do welfare the old persons needs the hand and the labor. Therefore, the hand and the labor can be reduced by using the robot for welfare. I think of the aging society to the future and using Welfare Robot Prototype of the two wheels drive, I controlled in the running. It makes programming the running control of me to run safely and moreover smoothly and it mistakes it in the attempt from the gotten data. Then, the main system of Welfare Robot Prototype that was used by the running experiment this time is shown below.

2. Introduction

The Welfare Robot Prototype is a mobile robot has four wheels; two wheels, located along the central axis, are driven wheels, and the other two are auxiliary wheels to support the body of the mobile vehicle. A pair of Direct Current (DC) motors turns the driven wheels, such that the speeds of the two DC motors (right and left) may be changed independently, via the Input-and/or-output (I/O) card, Digital-to-Analog (D/A) converter, and a Pulse-width-Modulation (PWM) controller card. The relative position or motion of each on the shafts. Power is supplied by DC batteries installed in the body. There are 6 ultra sonic sensors facing forward and rear for using in obstacle detection, which may be read via the input ports on the I/O card. Simple outputs may be obtained via 4 Lights Emitting Diodes (LEDs) mounted in the head . For more complex interaction, the robot has a touch panel Liquid Crystal Display (LCD) monitor. The robot has two color Charge Coupled Device (CCD) cameras mounted on its head. Each camera is capable of independent pan and title movement. The head can be rotated right and left using a stepping motor. Connections also exist for speaker output and microphone throughout. To control the robot, there are two personal Computers (PCs) built into mentioned earlier (*I/O*, motor driven, frame grabber, and counter cards), and the serial ports (for camera control). The PCs have Windows 98, Microsoft Visual Studio (MSVS) development software, and hardware driver software installed. The two PCs may communication via Ethernet. Looking at the robot from the rear, the two PCs can be seen with their various connections. The proposed division of tasks between these two machines is as shown in Fig.1.1.

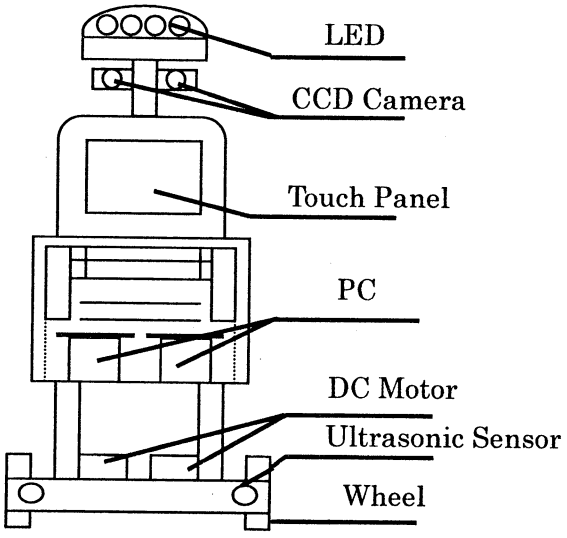
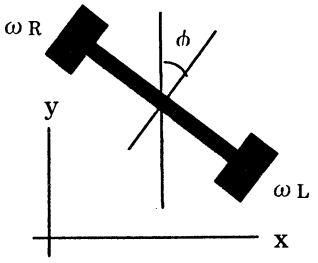


Fig.1.1 The Welfare Robot Prototype

3. Robot Wheel Geometry

Networked messages will carry information about items seen (or head) from one PC (Which has access to the frame grabber and sound card) to the PC responsible for controlling hardware. In order to interpret any information retrieved from the camera image, it is necessary to know the basic geometry of the robot. Robot wheel geometry, and motor equations are displayed in Figure 1.2.



Position(x,y, ϕ)
Wheel Speed ω
Wheel diameter
d=100(mm)
Wheel distance
D=400(mm)

$$\begin{cases} \phi = \pi\ (\omega_L - \omega_R)\ d\ /\ D \\ x = 0.5\ (\omega_L + \omega_R)\ \pi\ d \sin(\phi) \\ y = 0.5\ (\omega_L + \omega_R)\ \pi\ d \cos(\phi) \end{cases}$$

Fig. 1.2 Robot Motion

The position of the robot at any time may be expressed as a pair of co-ordinates (x,y) and an orientation angle (ϕ). The rate at which the position (x,y,ϕ) changes is dependent on the sum and difference of the rotational speeds of the *left/right* wheels (ω_L, ω_R). The actual speed of the left and right wheels at any time may be estimated by either counting or timing the pulse from the left and right micro-sensors which detect the slots in the encoder disks as shown below:

$$\omega = \frac{2\pi}{n} \times \frac{P}{S} \quad (1)$$

for an encoder with n slots, where P pulse counted in a sampling time of s seconds,

$$\omega = \frac{2\pi}{n} \times \frac{1}{s} \times \frac{1}{i} \quad (2)$$

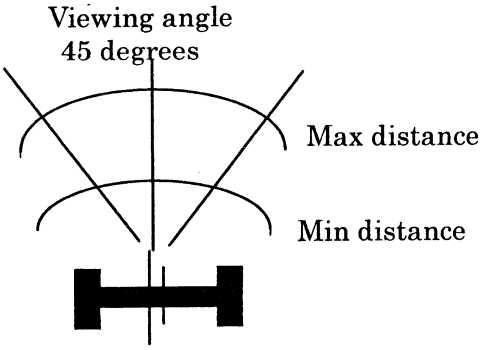
for an encoder with n slots, where I intervals of s seconds counted between pulses. The motors are controlled by switching the *on/off*, forward/reverse relays, and by sending a number d between 0 and 255 to the PWM controller via the *I/O* port and *A/D*. The speed of the motor (in either direction) under no-load is linearly related to the number. The empirically estimated relation is:

$$P = s(235 - 12.3 \times d) \quad (3)$$

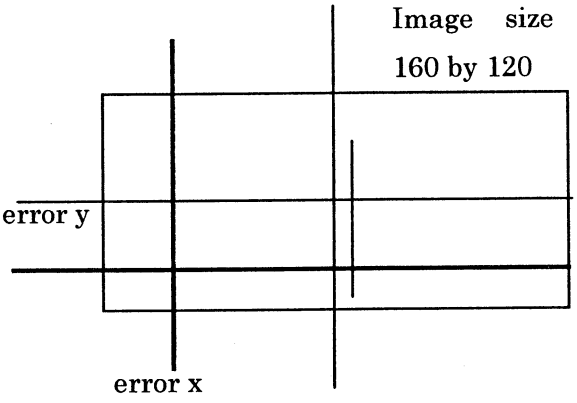
There is some slight difference between the motors that is changing over time and that this relation will change with loading. The time constant of the motors has been experimentally determined from non load step response, as 20 ~30ms. Practically this means that when a speed command is sent, the robot will take approximately 100ms to achieve that speed.

4. System of Camera

The location of any object in a single camera image is related to its position in the "real" world. The "error" is the vertical and horizontal offset of object image, from the center of camera image as shown in Fig.1.3.



Left camera axis is displaced from robot central axis by 0.08(m)
= 30pixel offset at min distance



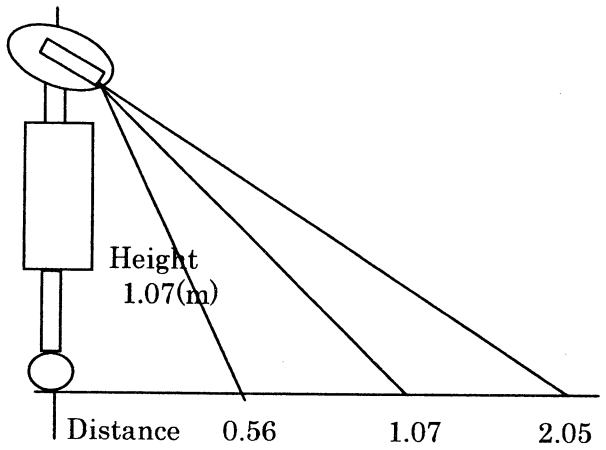


Fig. 1.3 Robot Geometry

This figure shows only the left camera geometry, relating object position to error, pixel offset and camera focal length. The distance to object and height of object and angle to object are not uniquely specified by a single camera image. If similar calculations may be made for the right camera, the right camera, the information from the two cameras may be used to uniquely determine these quantities.

The cameras and capture card are NTSC standard. NTSC frame rate is slightly less than 30 frames / second. Therefore, position information cannot be obtained from the image faster than this rate.

The both cameras provide color images. In order to understand the data that is obtained from the camera capture card, it is necessary to introduce the concepts of the pixel and of color space. The pixel is the smallest region that the image is sub-divided into. We can consider an image to be an array of pixels. Color space concerns the way in which a color can be represented or measured. It is useful to image color space as a co-ordinate system in which the colors are located, similar to the way that robot position can be represented in the x, y, ϕ system. Depending on which color space system is used, the amount of bytes required to represent a pixel's color will vary.

For each pixel, the color space information is made available from the capture card. This information is stored sequentially, row by row, starting from the top of the image. Regardless of the number of bytes used to represent the color the data provided for a signal row will always be a multiple of 4, if the row information ends before a 4 bytes boundary where the next row starts.

The color spaces typically used are Red, Green, Blue (RGB) orthogonal color space. Any color may be considered as a mix of three primary colors red, green and blue. This color space attempts to quantify how much of each primary color (between D and maximum), is in the color observed.

Hue, Saturation, Value (HSV) cylindrical color space. A color is defined by its hue which is expressed as an angle. The saturation indicates the purity of the color. The brightness or darkness of the color is the value.

Luminance, Chrominance (YUV) orthogonal colors space. The luminance Y of a color is the amount of brightness it contains. This is equivalent to the gray scale value of the color. The chrominance U,V may defined in many different ways, but reflect the proportions of two reference colors in the final color.

The cameras have a viewing angle of 45° about the center, and a pan range of $\pm 30°$. This when combined with the head rotation range of $\pm 100°$ gives the full viewing area as $\pm 152.5°$.

5. Motion of this Robot's Head

The head angle may be changed by instructing the motor card to send a specified number of pulses to the head stepping motor. The frequency with which the pulses are delivered varies in accordance with the specified motor acceleration parameters. In this way smooth motion can be achieved. Alternatively, the speed of a continuous motion may be specified and changed as the head moves. To control the camera pan angle (or any of the other camera parameters such as tilt, filtering etc) the SONY VISCA serial protocol may be employed, across the cable from the camera connected to the PC serial port. This is an asynchronous protocol meaning that messages may be sent at any time. A message consists of a header (indicates who sent the message, what type of message), message and

terminator. A message sent from the computer that meets the specified format will be acknowledged, and answered if it is a query, or executed if it is a command. In the case of commands, a second acknowledgement is sent when execution is completed.

Network communication may also be implemented on an asynchronous basis, between computers. In order to do so, one computer must act as the server, and the other as the client. At startup, the client.request that the server assign it a socket, across which they send messages to each other. While it is not necessary to have a header and terminator on these messages, such things facilitate relative communication.

As well as visual information, the network will also have to carry voice command or sound information between the two computers. While second is based time as is the camera image, it must be analyzed in a different manner as no signal instant in time may be considered separately from all the others (unlike for the camera where we can consider a signal image). Sound is a waveform, and as such may be represented by amplitude through time, frequency spectrum, or any waveform characteristic, for example the Discrete Fourier Transform (DFT).

In Addition to using sound to interrupt voice commands, it is also used to detect obstacles. The ultrasonic sensors consist of a sender and receiver. Sound waves from the sender, bounce off objects and are detected on reflection. The sensors have a narrow beam angle, and their detection distance can be adjusted, by adjusting the sensitivity of the receiver.

The many parts of the robot need to be controlled simultaneously. In order to do so effectively under the Windows 98 operating system, it is necessary to use multiple processes and multiple threads. A process can split its work between threads that it starts and can subsequently control. Multiple threads can have access to the same memory space, Multiple processes each have their own memory and code space, and are used for unrelated tasks. They have no access to shared memory space with another process, without some explicit definition as such.

This completed the basic concepts involved in understanding and controlling the robot

6. Conclusions

To the Future, we will study to control the rotational numbers. The speed is the optimal to make run safely and smoothly must be done.

Next, we will do the pursuit control what runs by chasing the color tape that was pasted to the floor by two CCD cameras what has loaded into the robot. To suppress an error with the goal one, when the robot is bending, a lot of problems to have removed a line mainly, and so on, seem to occur about the curve.

We search into the cause and think that must solve one by one.

7. Reference

1) M. Sugisaka: Beginner system theory and control P83～120(1997)

2) An explanatory note of The Welfare Robot Prototype P4～8(2000)

3) H. Hayashi: New Visual C++ 6.0 for BeginnerP3～204(1998)

4) Borland International: Programmers guide Borland C++ 5.0 P5～352(1996)

5) H. Hayashi: Revision New C Language for beginner P2～384(1998)

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Periodic Cluster Attractors and their Stabilities in the Turbulent Globally Coupled Map Lattice

Tokuzo Shimada and Kengo Kikuchi

Department of Physics, Meiji University, Higashi-Mita 1-1-1, Kawasaki, Kanagawa 214-8571, Japan

Abstract

The Globally Coupled Map Lattice (GCML) is one of the basic model of the intelligence activity. We report that, in its so-called turbulent regime, periodic windows of the element maps foliate and systematically control the dynamics of the model. We have found various cluster attractors. In one type of them, the maps split into several almost equally populated clusters and the clusters mutually oscillate with a period p that is the same with the number of clusters (c). We name them as maximally symmetric cluster attractors (MSCA's). The most outstanding is the *p3c3* MSCA and its bifurcate. The MSCA is proved to be linearly stable by Lyapunov analysis. There are also cluster attractors with $p>c$. They come out in sequences with increasing coupling. The formation of the clustors in the very weakly coupled chaotic system may suggest a new form of an intelligence activity.

1. Introduction

We have recently found that, in the so-called turbulent regime of the GCML, a variety of amazing periodic cluster attractors are formed even though the coupling between the element maps is set to be very small [1]. In this note we review our work and substantiate it by ample examples. For a pioneering work of this model, see e.g. Ref. [2].

The simplest GCML—a homogeneous one—is defined by

$$x_i(t+1) = (1-\varepsilon) f_a(x_i(t)) + \varepsilon h(t), \tag{1}$$

$$h(t) = \sum f_a(x_j(t))/N, \tag{2}$$

with $f_a(x) \equiv 1 - ax^2$. All maps are endowed with a common high nonlinearity and evolve under an averaging interaction via their mean field $h(t)$ with a coupling ε. The formation of clusters via synchronization in this model is well studied for the large coupling region and the switch between coded attractors has been investigated in detail. For a recent progress, see [3]. On the other hand, in the small coupling (so called turbulent) region, maps have been regarded to evolve almost randomly under some hidden-coherence. Our new observations of various cluster formation in this region indicate that a pattern-recognition in a complex system is possible even if a highly random system is set with a very weak coupling.

We establish the linear stability of the cluster attractors formed in the turbulent regime. We in particular derive algebraically the ε value for the formation of the MSCA.

2. Foliation of the Element Map Windows

In a MSCA configuration the mean field must be time-independent due to a high symmetry in the cluster populations. Thus (1) becomes

$$x_i(t+1) = (1-\varepsilon) f_a(x_i(t)) + \varepsilon h^*, \tag{3}$$

with a constant external field h^*. The evolution equation acts on all the maps commonly in (1) and it becomes furthermore independent of time in (2). We can cast this unique equation into a standard logistic map with a reduced nonlinear parameter b,

$$y_i(t+1) = 1 - b(y_i(t))^2, \tag{4}$$

by a linear scale transformation

$$y_i(t) = (1 - \varepsilon + \varepsilon h^*)^{-1} x_i(t), \tag{5}$$

and the reduction factor of nonlinearity is

$$r \equiv b/a = (1-\varepsilon)(1-\varepsilon(1-h^*)). \tag{6}$$

The clusters of MSCA oscillate mutually around the fixed average h^*. Their orbits are the same each other *modulo time translation* and propor-

tional to the orbit of a logistic map with the nonlinearity b reduced by a factor in (6). Thus it must hold that

$$y^* = (1 - \varepsilon + \varepsilon h^*)^{-1} h^*, \tag{7}$$

with $y*$ being the time-average of the logistic map at b. Equation (6) and (7) are the key to find how a periodic window of an element map foliates and produces a MSCA of GCML. On one hand there is a periodic window at b with $y*(b)$. On the other hand there is a MSCA produced in GCML at a, ε with a constant mean field $h*$. The nonlinearity a of the latter element map is reduced to b by the averaging interaction. For each reduction factor r, we can work out a and ε that corresponds to b by eliminating $h*$,

$$a^{(b)}(r) = b/r, \qquad \varepsilon^{(b)}(r) = 1 - ry^*(b)/2 - \sqrt{r(1-y^*(b)) + (ry^*(b)/2)^2}. \tag{8}$$

Now by varying r, (8) gives *a curve of balance* on the a,ε-plane, which emanates from the point $(b,0)$. We call this curve as ***a foliation curve*** of a window dynamics. *If a MSCA with a period p is to be produced, it must be produce in a GCML with the parameter a, ε set on the curve of the period p window. At some stronger coupling at given a, the maps should be more tightly bunched and we may expect p > c type attractors.*

Let us check if this prediction works. For this purpose we consider the mean squared deviation (MSD) of the mean field in time

$$\langle \delta h^2 \rangle_T \equiv \Sigma_t (h(t) - \langle h \rangle_T)^2 / T, \tag{9}$$

as an indicator of the cluster formation. A MSCA will yield very low MSD due to its high symmetry in cluster populations, while the $p > c$ attractors will give remarkable peaks due to the lack of one or more clusters. In Fig.1 each panel is set at a fixed a and MSD is shown as a function of ε for $N=10^4$ GCML. Six prominent logistic widows ($p = 7, 5, 7, 3, 5, 4$ with increasing b —decreasing ε at the same a) are selected and the family of foliation curves of these windows is shown underneath the panels. The four curves A-D for each window respectively come from the point A below the threshold, the threshold B, the first bifurcation point in the window C, and the closing point D. The shaded zones in each panel are then the expected place of the manifestation of prominent windows. At each zone, a MSD valley due to MSCA should appear in the lower ε side and a MSD peak by $p > c$ cluster attractors at the nearby higher ε. We find that the prediction works with almost no failure in all panels and in all six windows.

Interestingly, the MSD curve in each panel has an ample amount of peaks and valleys at the smaller ε region (the left), but only a few broad ones at the larger ε. This is naturally understood as follows. In a way, *each panel is a screen which displays the windows of the single logistic map by using a macroscopic coherent state of GCML.* But the panels are inclined; a smaller ε implies less reduction, i.e. $r \approx 1$. Hence the left sensitively displays the sharp peak-valley structure induced by cluster attractors. The right, on the other hand, can reflect only the accumulation of the periodicity remnants from nearby windows, being dominated by the prominent one at its respective zone.

We have checked that, at all the MSD valleys with large nonlinearity reduction, the $h(t)$ distribution is Gaussian with the MSD sizably larger than the value dictated by the law of large numbers—the so-called hidden coherence [2]. *We therefore conjecture that the hidden coherence is due to a desynchronized MSCA state* [1].

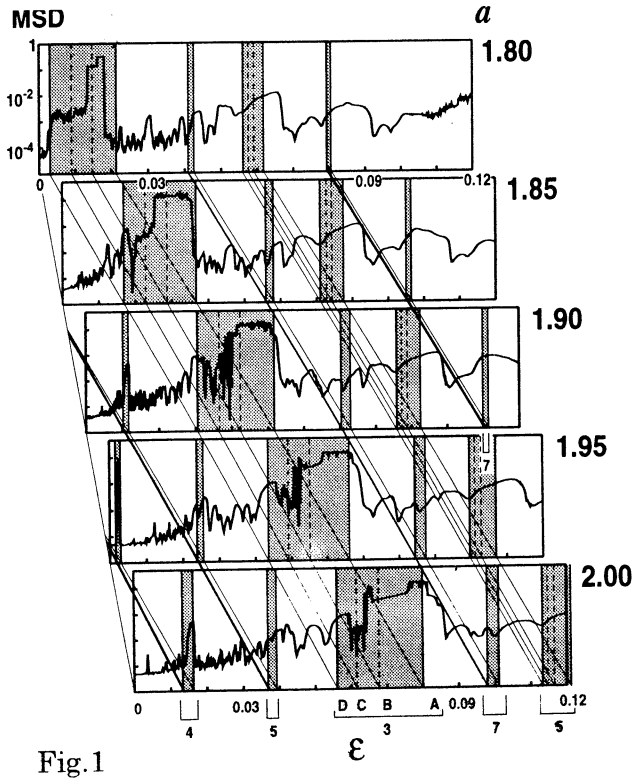


Fig.1

3. Lyapunov Stability Analysis

The Lyapunov analysis can be applied to both diverging and converging system orbits and it can detect the possible coexistence of multifold finial states. It tracks the expansion rate of a shift vector under the linearized GCML equation

$$\delta x_i(t+1) = -2a[(1-\varepsilon+\frac{\varepsilon}{N})x_i(t)\delta x_i(t) + \frac{\varepsilon}{N}\sum_{i\neq j} x_j(t)\delta x_j(t)], \quad (10)$$

and yield the maximum exponent λ_{max}.

First let us investigate the outstanding period three cluster attractors. In Fig.2 we compare λ_{max} and the MSD. $N=10^6$, $a=1.90$.

We find three salient structures.

(1) MSCA*. *A seagull structure (ε=0.032-0.037)* in both. *For the most stable events, the MSD is also the least.* By a direct observation of the orbits, we find all events are bifurcated MSCA. This can be understood as follows. The maps and the mean field together are a bootstrap system, see (1). That is, generally the mean field is not a simple external source and the fluctuation in it will be reflected to the fluctuation of the maps. The high mean field fluctuation would lead to the instability of the system. An exception is the MSCA. Here the mean field is constant and the system protects itself from instability which is otherwise amplified by the bootstrap. *The MSCA is the configuration with which the GCML stabilizes itself with minimum fluctuation of the mean field.*

(2) p3c3 MSCA. *The first low band (0.037-0.041).* The negative λ_{max} and the low MSD.

(3) p3c2 cluster attractor. *The second low band (0.041-0.051).* Here, the MSD is extremely high because of a lack of one cluster to minimize the fluctuation. For the bulk of events we find λ_{max} is small but positive. For a system with low degrees of freedom, the positive λ_{max} implies chaos. But here, even with a positive λ_{max}, the maps always form stable *p3c2* state. There is actually no contradiction. The global motion of the clusters is periodic, but, inside each cluster, maps are evolving randomly. The Lyapunov exponent is sensitive to the microscopic motion and hence yields positive value. But for a larger deviation, nonlinear terms can become relevant and pull back the map. *This type of map motion—microscopically chaotic but macroscopically in the periodic clusters—may be called as confined chaos.*

Note that in Fig.2 there are also states with high rate mixing of maps (denoted as M), which coexist with the cluster attractors at the same ε.

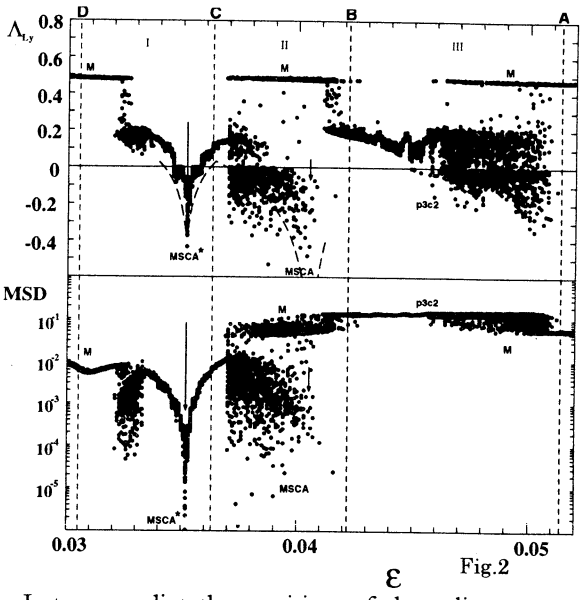


Fig.2

Let us predict the position of the salient cusp. For a cluster attractor configuration, there occurs a high degeneracy of eigenvalues of the linear stability matrix. For each cluster (I) with N_I maps, there is a single eigenvalue with (N_I-1)-fold degeneracy. It is

$$\Lambda^{(I)} = (-2a(1-\varepsilon))^p \Pi_{k=1}^p X_k^I, \quad (11)$$

where X^I denotes the cluster orbit. It is responsible for the stability of a map in that cluster. The altogether c eigenvalues of this type take care of $\Sigma_{I=1}^c (N_I - 1) = N - c$ degrees of freedom of maps. The other c eigenvalues are responsible for the stability of the cluster orbits. For a MSCA the product of X_k^I over the period p is common to I. Therefore the spectrum consists of a highly degenerate eigenvalue Λ with $(N-c)$-fold degeneracy and additional c non-degenerate eigenvalues.

Now, the crucial point. The orbit X_k^I of a MSCA cluster at a, ε (common to all I) is nothing but the orbit of a single map y_k at b modulo a scale factor (see (5)). *Thus, Λ for MSCA agrees with the Lyapunov exponent of a single map at b;*

$$\Lambda = (-2b)^p \Pi_{k=1}^p y_k. \quad (12)$$

This becomes zero when one of the orbit points y_k's becomes zero, that is, when b is a solution of

$(f_b)^p(0)=0$. As for the other c eigenvalues, they agree with Λ within small correction of order ε/c [1]. That is, if $\lambda_{\max}=\log(|\Lambda|)/p$ is $-\infty$, they are all extremely small, approximately $\log(\varepsilon/c)$. For *p6c6* cusp, the relevant solution b_{c6} is 1.77289. The foliation curve for this b reaches the $a=1.90$ panel at ε_{c6}=0.0352. This is precisely the MSCA* cusp position. The predicted curve for $\lambda_{\max}$ around the cusp (the dashed line) is also in good agreement with the data.

Now let us investigate the cluster attractors with higher periodicities.

p5 and p4 cluster attractors

In Fig.3a we show the same with Fig.2 for the foliation of the *p*5 window—the sequence of *MSCA**→*c5*→*c4*→*c3*→*c2* sampled at $a=1.64$ (left). Fig.3b is for *p4; MSCA**→*c4*→*c3*→*c2* at $a=1.95$. Both for N=10^4. Overall agreement with the *p*3 case can be seen clearly—we observe the MSCA cusps in D-B and $p>c$ clusters in B-A. Algebraically we obtain b_{c10}=1.62943 and b_{c8}=1.94178. The predicted MSCA* cusp positions from our foliation equations are ε_{c10}=0.00397 and ε_{c8}=0.00194 —in agreement with the observed ones.

In Fig.4 we show the composition of $p>c$ attractors obtained by a gap analysis of 10^3 random events at each ε for each of *p*5 and *p*4. With increasing ε, the c sequentially decreases with intermediate coexisting phases of $p>c$ clusters and random maps. The MSCA dynamics dominates inside the window (above the dashed line B) while the $p>c$ clusters are formed in the intermittent region below the window (the region of the higher reduction by ε).

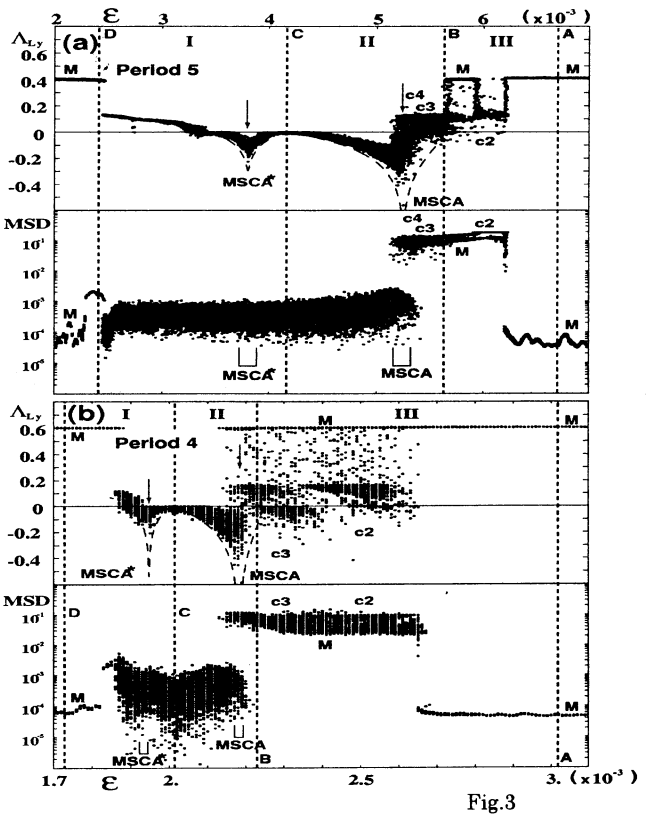


Fig.3

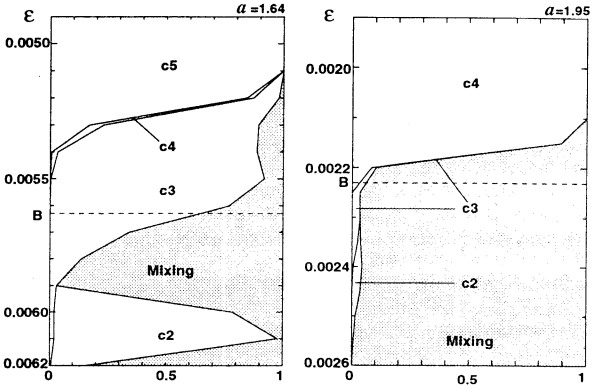


Fig.4

4. Conclusion

The newly found stable self-organized MSCA's (and its bifurcates) with a minimum fluctuation are the basic states of maps in the turbulent GCML. They may be regarded as counter parts of the ordered vacuum in the field theory at the spontaneously broken symmetry phase. The $p>c$ attractors are curious deformed states at slightly higher coupling. Their periodic orbits are almost the same with the MSCA but due to the lack of some clusters the MSD is maximized. Even when the Lyapunov exponent is positive, the maps are macroscopically confined in clusters stably.

The foliation has been also found independently by two other groups [4] but neither the stability nor the $p>c$ attractors were discussed by them.

---

[1] T. Shimada, Tech. Rep. IEICE, NPL97-159, 71(1998); T. Shimada and K. Kikuchi, Phys. Rev. **E**62, 3489 (2000).

[2] K. Kaneko, Phys.Rev.Lett. **63**, 219 (1989); ibid. **65**, 1391 (1990).

[3] T. Shimada and S. Tsukada, *A resolution of the puzzle of the posi-nega switch mechanism in the globally coupled map lattice*, submitted to this conference (AROB01-6).

[4] T. Shibata and K.Kaneko, Physica D **124**, 177 (1998). A.P. Parravano and M. G. Cosenza, Int. J.Bifurcation Chaos **9**, 2331(1999).

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Construction of Inverse Model for Data Mining by using Probabilistic Neural Network

K. OKUHARA†, H. FUJITA and T. TANAKA†

† Dept. of Management and Information Sciences,
Hiroshima Prefecture University,
562, Nanatsuka, Shyobara 727-0023, Japan.

Abstract

In this paper we propose an application of a neural network to a decision support system. To estimate an unknown nonlinear model, the probabilistic neural network is incorporated into the proposed system. We further propose the survival of the fittest type learning rule which can be applied to the update rule of the probabilistic neural network. Our model is an algorithm which extends the deterministic annealing Expectation Maximization algorithm. It can execute faster learning and avoid over-fitting rather than the usual probabilistic neural network. The effectiveness of the decision support system using such learning rule is shown for the sampling data.

1 Introduction

The multiple regression is one of method to show the relationship between the explained variable and some explanatory variables. It has however the possibility that a large residual is left, because of such modelling assuming the linear transformation from the explanatory variables to the explained variable. Furthermore from the results of linear analysis, we can only detect the general tendencies. Compared with the linear modelling, the nonlinear modelling can not only reduce the residual but also advise a detailed policies in response to a situation. Such features are very important.

In this paper, we propose a decision support system. We apply a neural network which can estimate an unknown nonlinear model by learning to the proposed system. Recently, it is known that the neural networks are available for the pattern recognition, the signal processing and so on[1], various neural network models have been proposed according to cases. For our purpose, we select the probabilistic neural network[2] which can reconstruct the joint probability density function of the explained variable and some explanatory variables. On constructing our system, we improve the probabilistic neural network by considering the survival of the fittest type learning rule[3]. The neural networks applying our proposed learning rule can execute faster learning and avoid the over-fitting.

2 Proposed System

2. 1 Structure

In this study we use a Probabilistic Neural Network (PNN)[2,4] for proposed decision support system. PNN is one of NN which estimates an unknown probability density function $p(\mathbf{z})$ by summing up many outputs of Radial Basis Function (RBF). A normal probability density function is used for the RBF. Figure 1 shows the structure of PNN which consists of the K input neuron and 1 output neuron.

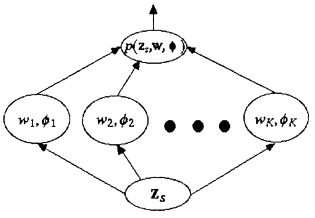


Figure 1 The structure of the PNN

An input vector $\mathbf{z}_s$ can be described by $\mathbf{z}_s^{\mathrm{T}} = [\mathbf{x}_s^{\mathrm{T}} \mathbf{y}_s^{\mathrm{T}}] \in \Re^{(N+M)}$ where $\mathbf{x}_s = [x_{\mathbf{s}}^1, x_{\mathbf{s}}^2, \cdots, x_{\mathbf{s}}^N]^{\mathrm{T}} \in R^N$ and $\mathbf{y}_s = [y_{\mathbf{s}}^1, y_{\mathbf{s}}^2, \cdots, y_{\mathbf{s}}^M]^{\mathrm{T}} \in R^M$. Supposing that the relation between the vector $\mathbf{x}_s$ and $\mathbf{y}_s$ in the RBF of the input neuron

$$N_{(N+M)}(\mathbf{z}_s, \phi_k) = N_N(\mathbf{x}_s, \phi_k^{\mathbf{x}}) \times N_M(\mathbf{y}_s, \phi_k^{\mathbf{y}}) \quad (1)$$

where

$$N_{(N+M)}(\mathbf{z}_s, \phi_k) = \frac{1}{(2\pi)^{(N+M)/2} ||\mathbf{\Sigma}_k||^{1/2}}$$

$$\times \exp\left\{-\frac{1}{2}(\mathbf{z}_s - \mathbf{m}_k)^{\mathrm{T}}\boldsymbol{\Sigma}_k^{-1}(\mathbf{z}_s - \mathbf{m}_k)\right\}. \quad (2)$$

T shows a transpose of matrix, $\mathbf{m}_k$ is a mean vector $[m_k^1, m_k^2, \cdots, m_k^{(N+M)}]^{\mathrm{T}} \in \Re^{(N+M)}$ and $\boldsymbol{\Sigma}_k^{-1} \in \Re^{(N+M)} \times \Re^{(N+M)}$ is an inverse matrix of the covariance matrix $\boldsymbol{\Sigma}_k$ whose ijth component is σ_k^{ij} ($i, j = 1, 2, \cdots, N+M$). This means that each input neuron can consist of cells which receive $\mathbf{x}_s$ and cells which receive $\mathbf{y}_s$.

This result say that we can consider the right output neuron group which output $R_{ij} = \mathrm{E}_{\mathbf{R}}[x_s^i|\mathbf{z}'_s : \theta]$, $(i = 1, 2, \cdots, N;\ j = 1, 2, \cdots, M)$ and the left output neuron group which output $L_{ji} = \mathrm{E}_{\mathbf{L}}[\mathbf{z}'_s|x_s^i : \theta]$ where $\mathbf{z}'_s = [x_s^1, \cdots, x_s^{i-1}, x_s^{i+1}, x_s^N, y_s^1, \cdots, y_s^j, \cdots, y_s^M]^{\mathrm{T}} \in \Re^{(N+M-1)}$ (See Figure 2).

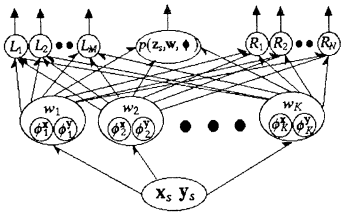


Figure 2 The structure of the proposed system

2. 2 Dynamics

In figure 2 $\mathbf{w}$ denotes the set $\{w_1, w_2, \cdots, w_K\}$, ϕ denotes the set $\{\phi_1, \phi_2, \cdots, \phi_K\}$ and θ the set $\{\mathbf{w}, \phi\}$. For example the output which belong to the right output neuron group are given by

$$\begin{aligned}\mathrm{E}_{\mathbf{R}}[x_s^i|\mathbf{z}'_s : \theta] &= \int_R x_i p(x_i|\mathbf{z}' : \theta)dx_i \\ &= \sum_{k=1}^{K} \alpha_k(\mathbf{z}'_s) m_k^i \end{aligned} \quad (3)$$

where

$$\alpha_k(\mathbf{z}'_s) = \frac{w_k N_{(N+M-1)}(\mathbf{z}'_s, \phi^{\mathbf{z}'_k})}{\sum_{k=1}^{K} w_k N_{(N+M-1)}(\mathbf{z}'_s, \phi^{\mathbf{z}'_k})}. \quad (4)$$

$\phi^{\mathbf{z}'_k}$ means a set of the parameters concerned with $\mathbf{z}'_s$. The output which belong to the left output neuron group can be also obtained by the similar way.

The kth input neuron outputs for the input vector $\mathbf{z}_s$

$$\xi(\mathbf{z}_s, w_k, \phi_k) = w_k N_{(N+M)}(\mathbf{z}_s, \phi_k). \quad (5)$$

The output value $\xi(\mathbf{z}_s, w_k, \phi_k)$ is transmitted to the central output neuron, and central output neuron calculates

$$p(\mathbf{z}_s, \mathbf{w}, \phi) = \frac{1}{\sum_{k=1}^{K} w_k} \sum_{k=1}^{K} \xi(\mathbf{z}_k, \phi_k). \quad (6)$$

2. 3 Learning Rules

The parameter ϕ can be derived by maximizing the log-likelihood function

$$L(\mathbf{z}_s, \mathbf{w}, \phi) = \log p(\mathbf{z}_s, \mathbf{w}, \phi). \quad (7)$$

The each estimatee value of $\mathbf{w}$, $\mathbf{m}_k$ and $\boldsymbol{\Sigma}_k$ can be obtained by the iterative calculation derived by the Deterministic Annealing Expectation Maximization (DAEM) algorithm[5] which extends the EM algorithm[6]

$$w_k^{(t+1)} = \frac{1}{S}\sum_{s=1}^{S} h_k^{(t)}(\mathbf{z}_s), \quad (8)$$

$$m_k^{(t+1)} = \frac{\sum_{s=1}^{S} \mathbf{z}_s h_k^{(t)}(\mathbf{z}_s)}{\sum_{s=1}^{S} h_k^{(t)}(\mathbf{z}_s)}, \quad (9)$$

$$\boldsymbol{\Sigma}_k^{(t+1)} = \frac{\sum_{s=1}^{S} (\mathbf{z}_s - \mathbf{m}_k^{(t)})(\mathbf{z}_s - \mathbf{m}_k^{(t)})^{\mathrm{T}} h_k^{(t)}(\mathbf{z}_s)}{\sum_{s=1}^{S} h_k^{(t)}(\mathbf{z}_s)} \quad (10)$$

where

$$h_k^{(t)}(\mathbf{z}_s) = \frac{\left\{w_k^{(t)} N_{(N+M)}(\mathbf{z}_s, \phi_k^{(t)})\right\}^{\beta}}{\sum_{k=1}^{K}\left\{w_k^{(t)} N_{(N+M)}(\mathbf{z}_s, \phi_k^{(t)})\right\}^{\beta}}. \quad (11)$$

β denotes a positive constant. If the parameter β takes very small (≈ 0) then $h_k^{(t)}(\mathbf{z}_s)$ is regarded as the uniform distribution. This means that all $\mathbf{z}_s$ contribute uniformly to the estimation of the parameters. When β takes 1, the DAEM algorithm is equivalent to the EM algorithm. In the estimation of parameters, the DAEM can avoid the local optimal solution by using an annealing which increases β from a small value to 1 gradually.

2. 4 Models

We further improve the PNN to execute faster learning and avoid the over-fitting. The following survival of the fittest type learning rule is applied to the update rule[2] of parameter $\mathbf{w}$.

$$w_k^{(t+1)} = w_k^{(t)}\left\{\frac{1}{S}\sum_{s=1}^{S} h_k^{(t)}(\mathbf{z}_s) - w_k^{(t)} + 1\right\}. \quad (12)$$

This update rule works for pruning the redundant input neurons. This effect can realize the faster learning

and the avoidance of the over-fitting by using only necessary input neurons for estimating the unknown probability density function.We call such neural network a Competitive Probabilistic Neural Network (CPNN).

We can apply, of cause, the steepest decsent method to obtain the parameter θ which maximizes the log-likelihood function $L(\mathbf{z}_s, \theta)$, however it is easily considered that iteration number of learning will be fluctuation by the setting of learning rate. Therefor we apply the learning rule which is derived from the EM algorithm to the proposed system where the input neurons and the central output neuron is used for learning. Then the proposed system generates outputs of the right and left output neuron group by using obtained parameters. From reasons mentioned above, the proposed system is united model of the learning part and the output part through the input neurons.

In the simulation we first show the effectiveness of the proposed CPNN for nonlinear modelling. In order to show the effectiveness of the CPNN we consider the following three different learning algorithms for the PNN.

Model PNN: PNN using a EM algorithm which does not have a ability pruning neurons.

Model PNN+P: PNN using a EM algorithm which has a ability pruning neurons by an additional condition.

Model CPNN: PNN using a DAEM algorithm which has a ability pruning the redundant neurons by both the survival of the fittest type learning rule and an additional condition.

As the additional condition to pruning neurons, we consider that if the parameter w_k satisfies $w_k < C_w$ then the kth input neuron vanishes.

3 Simulation Results

3. 1 Effectiveness of CPNN

Table 1 shows the differences among these learning algorithms in applying to the same data. All models have 300 input neurons at initial state.

N_S denotes the number of surviving input neurons. The iteration number (I_N) of learning is counted at every update about all parameters w_k (k = 1,2,$\cdots$,K), m_k^i ($i = 1, 2, \cdots, N$) and σ_k^{ij} ($i, j = 1, 2, \cdots, N$).

From results obtained above and results of the multiple regression analysis, compared with the linear modelling, the nonlinear modelling can reduce the error E and the real error E_r. It is particularly known

Table 1 (a) Learning results ($C_w = 0.001$).

	N_S	I_N	$E(\times 10^2)$	$R_a^2(\%)$	E_r
PNN	200	4600	8.6	99.92	5448.96
PNN+P	116	2668	8.9	99.92	5493.42
CPNN	176	4048	4.4	99.98	3039.67

Table 1 (b) Learning results ($C_w = 0.003$).

	N_S	I_N	$E(\times 10^2)$	$R_a^2(\%)$	E_r
PNN+P	71	1633	11.9	99.85	6584.94
CPNN	135	3105	5.4	99.97	3072.24

that the model CPNN can estimate the joint probability density function of the sampling data better than the other two models PNN and PNN+P.

The number of survival input neurons (N_S) and the iteration number (I_N) of the model PNN+P are smaller than the model CPNN, which means that the usual PNN estimates the unknown function by using many redundant input neurons which generate small outputs. Such input neurons cause not only the delay of learning but also the over-fitting. However even if we eliminate such redundant input neurons by considering the condition of pruning, we can not expect a good performance for the nonlinear modelling by such model PNN+P which can not reduce a large error.

On the other hand, the proposed CPNN can reduce the errors and increase the adjusted coefficient of multiple determination R_a^2. The reason why it shows a good performance that the necessary input neurons to estimate a function tend to generate large outputs by the survival of the fittest type learning rule of the CPNN.

3. 2 Analysis Results by Proposed System

We secondly apply the proposed decision support system to the sampling data in agriculture from 1997 to 1999. In this case, $N = 11$ and $M = 1$. The relationship between the brix ($\equiv y_1$) and the other factors can be obtained by observing one of output $L_{1i} = \mathrm{E}_{\mathbf{L}}[\mathbf{z}'_s | x_s^i : \theta]$, ($i = 1, 2, \cdots, N$) of the left output neuron group after learning. Figure 3 shows the relationship between the leaf color ($\equiv x_3$) and the brix, where the real line output of the L_{13}.

We can easily understood that the proposed system could estimate the nonlinear transformation from the exogenous variables to the endogenous variable.

We finally shows that the proposed decision support system can advise a detailed policy in response

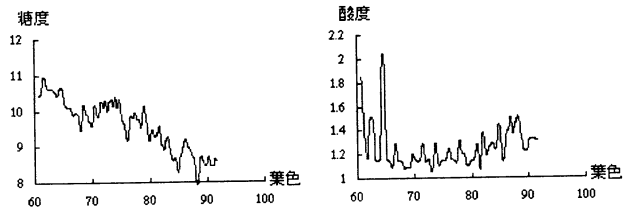


Figure 3 The relationship between the leaf color and the brix (or acid).

to a situation. In this case, it is assumed that the percentage of water content ($\equiv x_4$) is a controllable exogenous variable. Therefor we should observe the output $R_{41} = \mathrm{E}_{\mathbf{R}}[x_s^4|\mathbf{z}'_s : \theta]$. The inputs x_s^j $(j \neq i)$ are settled to sth sampling data when the system generates the output R_{41}. Figure 4 shows the analysis results for the 20th sampling datum where the real line output of the R_{41}.

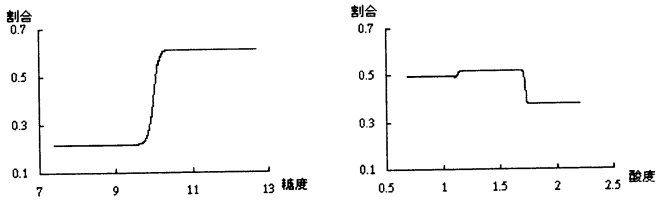


Figure 4 The maximum likelihood estimator of the percentage of water content [%] for the brix (or acid) at 20th sampling datum. (σ_k^{55} are given by learning.)

The line represents the maximum likelihood estimator. From figure 4, for example, the decision support system advises that the percentage of water content should be given by around 60% to make the brix up in the situation. That is, in order to increase the brix, the percentage of water content must be increased. However the results of this analysis show that there are some cases which are different from such tendencies in the past situations. The reason why we have such result which differs from the usual sense that it is stems from the consideration about the error of observation.

The proposed system can consider the error of the observation by settling the parameters σ_k^{55} $(k = 1, 2, \cdots, K)$. This means that we can estimate a large observation error for the CP by giving the parameters σ_k^{55} small values, which lets us be able to consider the existence of such latent error. Figure 4 (b) is a result obtained under the parameters σ_k^{55} are given by the small constants.

4 Summary

In this study, we propose an application of a neural network to a decision support system. We further propose the competitive probabilistic neural network applying a survival of the fittest type learning rule. Our model is an algorithm which extends the deterministic annealing Expectation Maximization algorithm.

In simulations, we show the effectiveness of the proposed CPNN for a nonlinear modelling, where the CPNN can can execute faster learning and avoid overfitting rather than the usual probabilistic neural network. Furthermore the results obtained by applying the proposed decision support system to the sampling data show that the proposed system can not only give the general tendencies which agree with the usual linear modelling but also advise the detailed policy which can not be gotten by the usual linear modelling by considering the observation error.

References

[1] D. E. Rumelhart, J. L. McClelland and the PDP Research Group, "*Parallel Distributed Processing*," **I**, MIT Press, 1986.

[2] R. L. Streit and T. E. Luginbuhl, "Maximum likelihood training of probabilistic neural networks," *IEEE Trans. NN*, **5**, No. 3, pp. 764-783, 1994.

[3] K. Okuhara, K. Sasaki and S. Osaki, "Reproductive and Competitive Radial Basis Function Networks Adapting to the Dynamical Environments," *Systems and Computers in Japan*, (to appear).

[4] K. Fukumizu and S. Watanabe, "The Statistical Inference Neural Network and Its Application to Pattern Recognition" *Tech. Report of IEICE.*, NC-92-36, pp. 83-90, 1992.

[5] S. Ueda and R. Nakano, "Deterministic annealing EM algorithm," *Neural Networks*, vol. 11, no. 2, pp. 271-282, 1998.

[6] A. P. Dempster, N. M. Laird and D. B. Rubin, "Maximum-likelihood from incomplete data via the EM algorithm," *J. Royal Statist. Soc. Ser. B*, **39**, pp. 1-38, 1977.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Function discovery system model using non-linear optimization method

Teruo SHIMOMURA*, Kazuki YAMASHITA**, Kazuhiro MORITA*** and Seiichi SERIKAWA****
Dept. of Electrical Eng., Kyushu Institute of Technology 1-1, Sensui-cho, Tobata-ku, Kitakyushu, Fukuoka 804-8550 Japan,
simomura@elcs.kyutech.ac.jp*, kazu@elcs.kyutech.ac.jp**,
morita@elcs.kyutech.ac.jp***, serikawa@elcs.kyutech.ac.jp****

Abstract In this paper, we propose a new function discovery system model (S-Poly model) for the artificial life type discovery system (S-System), by incorporating non-linear optimization technique in it. This model uses Polytope method for optimizing the values of the constants of the function under discovery process while evaluating the fitness of it with the observation data. The new model significantly reduces the number of generation required for the discovery. Also, we propose two other models combining Polytope with Genetic Programming (GP-Poly model) and mutation (Mut-Poly model). All these models are found effective in the function discovery, while S-Poly model is found to be the fastest among the three.
Key words: artificial life, genetic programming, function discovery, Polytope method

1. Introduction

J. Koza[1] first proposed the function discovery system based on Genetic Programming (GP). However, the system has some disadvantages. We thus proposed a bug type of artificial life based system to overcome these problems[2]. We called the system as S-System to indicate the incorporation of the concept of sexual and asexual reproduction in it. We then proposed a method to improve the search ability[3]. However, the system still needed considerably long time for the discovery. Hence, in this paper, we propose a non-linear optimization method incorporated S-System that significantly reduces the time for the discovery.

2. Algorithm of function-discovery

2.1 Main routine

Figure 1 is the flowchart of the algorithm of function-discovery by the use of a bug type of A-life proposed by us[2]. The flow is summarized as follows.

(1) Numerous bugs with the arbitrary function are generated at random. The number ***Pop*** is selected from the numerous bugs in order of high fitness.
(2) The generation ***Gene*** of the bug is set to 0.
(3) The value of the internal energy, $energy_p$, of all the bugs is initialized to 0.
(4) The procedures from (5) to (7) are repeated for all the bugs; the bug number ranges from 1 to ***Pop***.
(5) The bug p moves. This means that the values

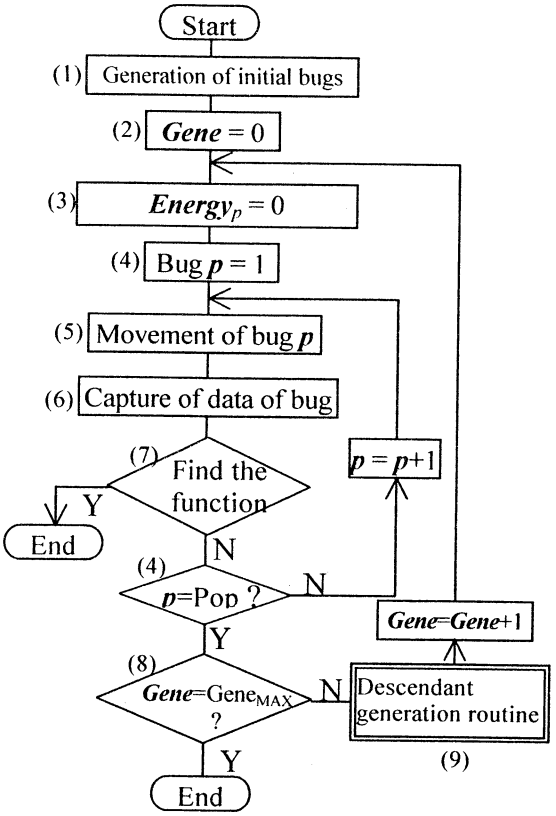


Fig. 1 The flowchart of the algorithm of function-discovery using the bug type of artificial life.

of constants $\overline{K}$ in the chromosome change slightly. That is to say, the values of K are replaced by $K \pm dK$, where dK is the small change of K, $K=(K_1, K_2, \cdots, K_n)$,

$\overline{\overline{dK}} = (dK_1, dK_2, \cdots, dK_n)$ and n is the number of constants in the chromosome. This concept is based on Ref. (4). The details are given in Ref. (2).

(6) The bug p catches the observation data (i.e., fitness fit_p of bug p is calculated from the observation data).
(7) In the case that fitness fit_p reaches the threshold fitness Fit_{TH}, this algorithm ends. This means a bug has discoverd the function f.
(8) The algorithm ends when the current generation ***Gene*** reaches the maximum Generation $Gene_{max}$.
(9) After the descendant-generation-routine is called, ***Gene*** is added to 1 and the algorithm returns to procedure (3).

2.1 Descendant-generation-routine

The flowchart is displayed in Fig. 2, and is summarized as follows.

(a) Based on the generation-gap, ***Par*** bugs are selected and they are passed down to the next generation. The elite strategy is adopted for the generation-gap.
(b) The bug number p is set to be 0. By the repetition of the following procedures from (c) to (g), ***Pop*** - ***Par*** bugs are generated.
(c) A bug is selected by the tournament strategy.
(d) The selected bug is judged whether it has the ability to sexually reproduce. In the case that the selected bug has the ability of sexual reproduction, procedure (e) is performed. In the other case, procedure (f) is carried out.
(e) The bug finds its partner, and they produce two children by crossover. Jump to procedure (g).
(f) Two children are produced by asexual reproduction.
(g) A part of the chromosome is changed by mutation at the rate of R_{mut}.

Thus, the descendants of the number of ***Pop*** are generated. For the details sexual/asexual reproduction and mutation, see Ref. (2).

3. Non-linear optimization method incorporated function discovery algorithm

GP and S-System are a kind of extension of Genetic Algorithm (GA) and uses the selection, crossover and mutation for function discovery. Table 1 shows how crossover and mutation change the function during the discovery process in GP and S-System. In both models, function type changes due to mutation. The difference between the two methods lies in the way of incorporating the crossover. In S-System, since the crossover is

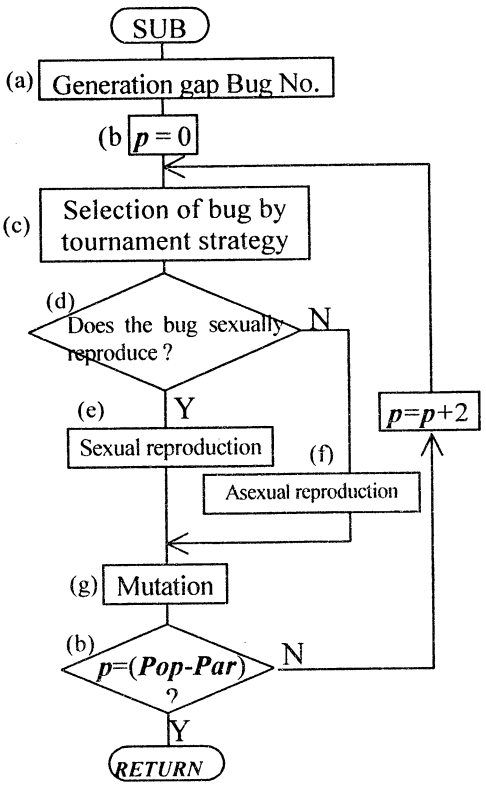


Fig. 2 The flowchart of descendant-generation routine.

Table 1 The change of function in GP and S-System.

Model	crossover	Mutation
GP	Function type*1)	Function type *1)
S-System	Values of constant only	Function type *1)

*1)Includes the condition where the function type does not change.

carried out only between the same type of functions, constants only change. Thus the schema does not get destroyed in S-System. This improves the local search ability[2]. However, the time taken for the discovery is long.

Thus, S-System is characterized with the evolution of the same type of functions. Moreover, we can think that if the function type is limited to one, the constants in it can be optimized by using one of the non-linear optimization methods[5]. From this, we incorporate a non-linear optimization method in S-System. We alter the previous S-System model in the following way.

(1) When the fitness is calculated, the constant $K(=k_1, k_2, \cdots)$ in the gene (function f) is.

changed using Polytope method[5] so as to minimize error. The new values of constant K' changed by the method is then replaced by K.

(2) The algorithm for movement of bug is removed.

(3) Although the target of crossover is a slight change of constant dK in the original model, we change it to constant K.

The other part of the algorithm of S-System is kept intact. Concerning non-linear optimization methods, we adopt Polytope method[5]. The reason is as follows: When there are many kinds of bugs, which have different type of functions, the evaluation of the gradient for all functions is impractical. Polytope method does not require the gradient function to be evaluated and hence we adopt this method.

The concept of movement, in which the value of constant change slightly[2], is not required in the new model, because the value of constant is changed by the Polytope method.

In general, the values obtained by Polytope method depend on initial values. In this model, the constant K in the gene (function f) corresponds to the initial values. This might results in a local minimum solution. In S-System, there exist many of functions with the different values of constant. This means that they have different initial values, so the probability of trapping in the local minimum is reduced. In addition, with the crossover in the above procedure (3), the values of constants (= initial values) can change greatly compared to the original model. This also helps to overcome local minimum trap.

4. Non-linear optimization method incorporated GP and mutation model

Table 2 Parameters for function-search.

No.	Parameters	Values
1	Population size	500
2	Initially created population size	100
3	Maximum generation	100
4	Generation gap	0.8
5	Tournament size	3
6	Crossover rate	0.5
7	Mutation rate of the chromosome	0.8
8	Mutation rate of the constant	0.1
9	Fitness for algorithm-end	0.96
10	Range of the constant by mutation	-10 ~ 10
11	Depth of nodes	8
12	Function set	+, −, ×, ^
13	Crossover type	Uniform

The new model is characterized with the change of the function type using mutation, with the optimization of the value of constants with Polytope method, and with the change of initial value using crossover. In the similar way, the following methods can also be thought.

(4) Remove (3) of above and make crossover between the different types of function. This is equivalent to combining GP model with the Polytope method. We call this as GP-Poly model.

(5) Remove (3) and (4) and make algorithm without crossover. This can be thought as mutation combined with Polytope method. We call this as Mut-Poly model.

5. Results

We now present the results of using the proposed models for function discovery. The conditions with which S-Poly model was used for discovery are shown in Table 2. For GP-Poly model, the item 8, 10, and 13 were removed and for Mut-Poly model item 5, 6, 8, 10 and 13 were removed. We used all the three models to search the Kepler's law that was performed in the previous paper[2]. All the models could discover the law at a few generations. Since the average of generations required by previous system to discover the law was about 300[3], we found the new models considerably faster. Computing time until discovery has been decreased to 1/20 or less. However, the difference in performance among the three models could not be understood due to an early discovery by all of them. Thus, we used the following expression to obtain dummy data sets for a complicated function type.

$$J = J_s \cdot \left\{ \exp\left(\frac{q \cdot V}{k_B \cdot T} \right) - 1 \right\} , \qquad (1)$$

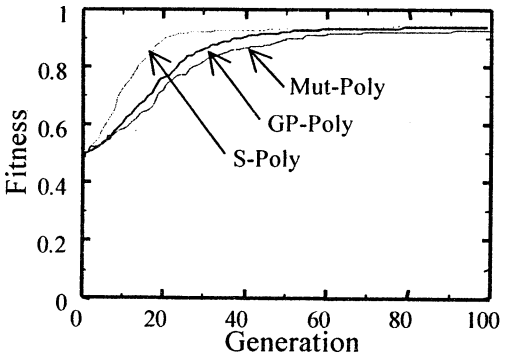


Fig. 3 The average of 30 experiments showing the relation between the generation and the fitness of the bug with the highest fitness.

where, $J_s = 1.0 \times 10^{-9}$, $q = 1.602 \times 10^{-19}$, $k_B = 1.382 \times 10^{-23}$. V was varied in steps of 0.01 from -0.05 to 0.1 in 16 steps and 3 value of T, 243, 273 and 303 were used to evaluate J. In total, 48 sets of data were generated as observation data. The three stated models were then used to search function $J=f(V,T)$. 30 experimental runs were performed. The discovery rate was found to be 0.80, 0.77 and 0.77 for S-Poly, GP-Poly and Mut-Poly models. Fitness over 0.96 was used as the condition for discovery. In Fig. 3, the transition of fitness that is an average of 30 experimental runs are shown. From the figure, it can be seen that all the models are fast in discovery. Among the three models, S-Poly model is the fastest.

Average computing times for each generation of S-Poly, GP-Poly and Mut-Poly models are 144, 180 and 115 seconds, respectively. Computing was performed with Pentium II 400 MHz processor operating on Windows NT system with C++ Builder 3.

6. Conclusions

In this paper, we proposed a new model called S-Poly model for function discovery by combining non-linear optimization method with the S-System. The proposed system could reduce the number of generations required for function discovery considerably. Also, by combining non-linear optimization method with genetic programming and mutation, we proposed GP-Poly and Mut-Poly models. All these models could discover the desired function effectively. The S-Poly model was found to be fastest on the average.

Asahi Technion Ltd has supported a part of this research.

References

(1) J. Koza, *Genetic Programming II*, *Auto Discovery of Reusable Subprograms*, *MIT Press*, p.109 (1994).

(2) S. Serikawa and T. Shimomura, "Proposal of a System of Function-Discovery Using a Bug Type of Artificial Life", *Trans. IEE*, Vol.118-C, 2, p.170, (1998).

(3) S. Serikawa and T. Shimomura, "Improvement of the Search Ability of S-System (A Function-Discovery System)", *Trans. IEE*, Vol.120-C, 8, p.1281, (2000).

(4) Iba, H., Higuchi T., de Garis, H, and Sato, T., "Evolutionary Learning Strategy using Bug-Based Search", *Proc. of the 13 Int. Joint Conf. on Artificial Intelligence (IJCAI-93)*, vol.2, pp.960-966 (1993).

(5) H. Okumura, *Algorithm dictionary using C language*, Gijutsu-hyoron, p.260, 1991, in Japanese.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Investigation of the Tritrophic System of an Ecological System by using an Abstract Chemistry

Yasuhiro Suzuki* Junji Takabayashi**
Hiroshi Tanaka*
*Medical Research Institute, Tokyo Medical and Dental University
1-5-45,Yushima Bunkyo Tokyo
113-8501 JAPAN
suzuki@gentzen.mri.tmd.ac.jp, tanaka@cim.tmd.ac.jp,
**Lab. Ecological Information, Graduate School of Agriculture, Kyoto University
Kyoto
606-8502 JAPAN
junji@kais.kyoto-u.ac.jp

Abstract

We investigated the population dynamics of such tritrophic interactions using a model of Abstract chemistries, Abstract Rewriting system of Multi-Sets (ARMS) which is a new research method in Complex Systems and Artificial Life. In ARMS model, we regarded a tritrophic interaction mediated by herbivore-induced plant volatiles that attract carnivorous natural enemies of herbivores as chemical reactions of four reagents (plants, herbivores, carnivores and volatiles). The intensity of interactions between individuals corresponded to reaction speed in the ARMS model. We compared the case where plants produce herbivore-induced volatiles vs the case where they do not with the model. Further, by changing the reaction speed, we found that there was a case where herbivore-induced volatiles that attract carnivores resulted in the population increase of the herbivores. We will discuss several conditions in tritrophic interactions that differently affect the population density of plants, herbivores and carnivores based on the prediction by the ARMS model.

Keywords: Ecological System, Artificial Chemistiries, ARMS

1 Introduction

A phenomenon that plants respond to herbivore feeding activities by producing volatiles that in turn attract carnivores enemies of the herbivores has been reported recently[2, 4]. These volatiles are not the mere result of mechanical damage, but are produced by the plant as a specific response to herbivore damage.

In mathematical ecological studies concerning with the system, one notable study is that of, Sabelis and de Jong(1996)[6] who reported that, when herbivore-induced volatiles profitable for plants, the kinds of the volatiles become polymorphic within species. They use game theory and show that ESS corresponds to the case when each species of plant produces the volatiles in polymorphic way.

In order to investigate the population dynamics of the tritrophic systems, we introduce an abstract rewriting system on multisets, Abstract Rewriting System on Multisets (ARMS) [7, 8]. This system is modeled as an *abstract chemical system* which is new research field in Artificial life and Complex systems.

In ARMS model, we regarded a tritrophic interaction mediated by herbivore-induced plant volatiles that attract carnivorous natural enemies of herbivores as chemical reactions of four reagents (plants, herbivores, carnivores and volatiles). The intensity of interactions between individuals corresponded to reaction speed in the ARMS model. We compared the case where plants produce herbivore-induced volatiles vs. the case where they do not with the model. Further, by changing the reaction speed, we found that there was a case where herbivore-induced volatiles that attract carnivores resulted in the population increase of the herbivores. In ARMS, a reaction rate is realized as the frequency of applying a rule.

2 *ARMS*

We will introduce the model, "Abstract Rewriting system on MultiSets (ARMS)." Intuitively, ARMS is like a chemical solution in which molecules floating on it can interact with each other according to reaction rules.

Technically, a chemical solution is a finite multiset of elements these elements correspond to molecules. A multiset allows to include the same elements, such as a,a,a,b,c,e. We can define the density by the number of symbols. For example, the density of "a" of the multiset is 3. Reaction rules that act on the molecules are specified in ARMS by such a rewriting rule as a,b $\rightarrow$ e,f. If the left hand side of the rule is included in the multiset, they are replaced by right hand side of the rule. For example, a,b $\rightarrow$ e,f rewrites $\{a, a, a, b, c, d\}$ to $\{a, a, e, f, c, d\}$. The reaction rate is expressed by the frequency of applying a rule.

In this section, we model the tritrophic system by using *ARMS*. We assume the symbol "a" as a leaf, "b" as a herbivore, "d" as a carnivorous and "c" as a certain density of herbivore-induced volatiles that attract carnivores, respectively. Furthermore, we add "e" as an "empty state" in order to introduce "death state." A plant is defined implicitly as the certain number of leaves. Evolution rule R_1 is defined as follows;

$$
\begin{aligned}
a &\xrightarrow{k_1} aa \quad r_1 \quad (\textit{increasing leaves}),\\
ab &\xrightarrow{k_2} bbc \quad r_2 \quad (\textit{a herbivore eats a leaf}),\\
dbc &\xrightarrow{k_3} dd \quad r_3 \quad (\textit{a carnivou catches herbivore}),\\
d &\xrightarrow{k_4} e \quad r_4 \quad (\textit{the death of a carnivous}),\\
b &\xrightarrow{k_5} e \quad r_5 \quad (\textit{the death of a herbivore}).
\end{aligned}
$$

where k_i illustrates the reaction rate. The r_1 corresponds to sprout and growth of a plant, r_2 to the case when a herbivore eats a leaf and the leaf generates volatiles, r_3 to a carnivorous catches herbivore, r_4 to the death of a carnivorous and r_5 to the death of a herbivore, respectively.

By using this model, we compared the case when leaves generate volatiles to does not. The evolution rules of the system without volatiles R_2 is defined as follows;

$$
\begin{aligned}
a &\xrightarrow{k_6} aa \quad k_6 \quad (\textit{increasing leaves}),\\
ab &\xrightarrow{k_7} bb \quad k_7 \quad (\textit{a herbivore eats a leaf}),\\
db &\xrightarrow{k_8} dd \quad k_8 \quad (\textit{a carnivou catches herbivore}),\\
d &\xrightarrow{k_9} e \quad k_9 \quad (\textit{the death of a carnivorous}),\\
b &\xrightarrow{k_{10}} e \quad k_{10} \quad (\textit{the death of a herbivore}).
\end{aligned}
$$

We set reaction rates of k_1, k_4, k_5, k_6, k_9 and k_{10} as 0.5, 0.1, 0.1, 0.5, 0.1 and 0.1, respectively.

3 Simulation and results

In both R_1 and R_2, the symbiotic relation consisting of plants, herbivore and carnivores (figure 3) can be found. It is interesting that R_2 (the system without volatiles) is likely to lose the relation under the same conditions (the number of plants, herbivores and carnivores in the initial state, reaction rate of r_3 and r_8).

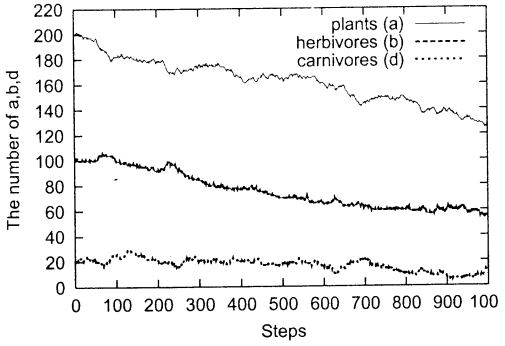


Figure 1: The symbiotic relation consisting of plants, herbivores and carnivores ($R_1, r_3 = 0.55$)

Figure 3 illustrates the symbiotic relation consisting of plants, herbivores and carnivores, R_1 is used and in the initial state, the population size of plants is 200, hervibore is 20, carnivores 10.

On the other hand, figure 3 illustrates that the case when R_2 is used under the same conditions as R_1 used. In this case, at first herbivores extinct thus carnivores can not survive.

The time course of population size of plants, herbivores and carnivores of R_1 (figure 3) is very different from R_2 (figure 3). In R_1 there are three attractors around (a, b, c) = (200, 100, 22)(1), (180, 105, 27)(2), (150, 120, 46)(3). At first the time course is attracted by (1) then (2) and finally (3). On the other hand in R_2 the course is attracted around (120, 200, 82), but departs from there in the long run.

These results show that the system with volatiles (plants produce volatiles) is more robust than the

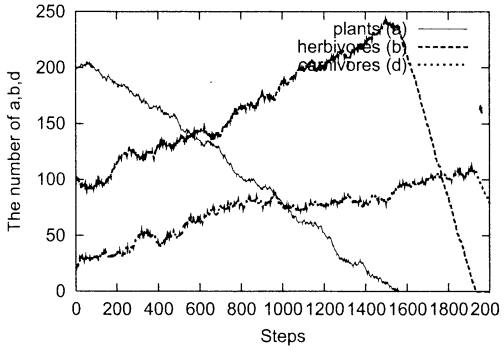


Figure 2: The case when the symbiotic relation is lost ($R_2, r_8 = 0.55$)

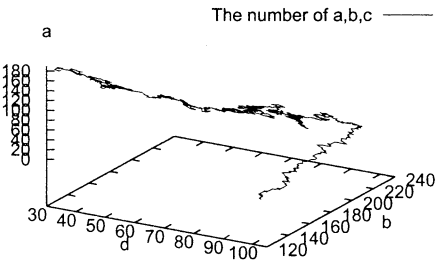


Figure 4: The time course of the population size of plants (a), herbivores (b) and carnivores (c) (until the plants extinct $R_2, r_8 = 0.55$)

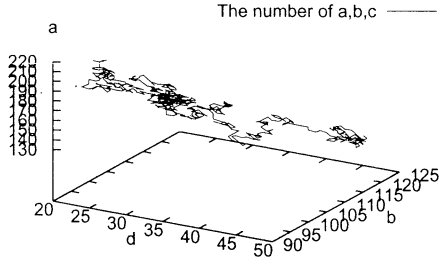


Figure 3: The time course of the population size of plants (a), herbivores (b) and carnivores (c) ($R_1, r_3 = 0.55$)

other. Thus, we are going to investigate the role of volatiles furthermore.

4 Discussion

Throughout the simulation, we discovered that herbivores could keep their population for at least 1000 generations in the system where the infested leaves generated the volatiles, whereas the herbivores were exterminated by the carnivores in around 100 generations in the system where leaves did not do so. In our model, the carnivores that use the volatiles to find the herbivores are not able to find their victims in a plant from which the volatiles were emitted under the detectable level for carnivores. This is probably the reason why the herbivores survived in the system where leaves generated the carnivore attractant.

This result suggests the possibility that herbivores induce the volatiles for their survival. This may be true in the tritrophic system consisting of plants, two-spotted spider mites (Tetranychus urticae) and predatory mites (Phytoseiulus persimilis) (Dicke et al. 1998). Two-spotted spider mite is a tiny (ca. 0.6 mm) herbivore. However, due to their rapid population increase on a plant, they tend to overexploit the plant. A kidney bean leaf infested by the spider mites started emitting volatiles that attract predatory mites P. persimilis (Maeda et al. 1998). Once in the prey colony, P. persimilis overexploit the spider mites. However, the volatiles were induced only after the spider mites increased over certain number per leaf (ca. more than 100-300 females per small plant) (Maeda et al. in prep) and there will be a time lag between the emission of the volatiles and the visitation by the predatory mites. Thus, the spider mites of the next generation that disperse from the current patch before the plant start emitting the volatiles can be free from the predator. Such spider mites will establish a new colony nearby. At the same time, the original colony may be exterminated by the predators. This cycle may be one of the defense strategies of the two-spotted spider mites against the predatory mites that search for them with the volatiles, and may be applicable for the prediction of the model. We will test this in the future experiments.

5 Conclusion

We proposed a new rewriting model, ARMS and applied it in order to investigate an ecological system that plants respond to herbivore feeding activity by producing volatiles that in turn attract carnivorous natural enemies of the herbivores. Throughout a simulation we obtain an interesting result that the possibility that herbivores induce the volatiles for their survival.

References

[1] Berry, G. and G. Boudol. 1992. The chemical abstract machine. *Theoretical Computer Science* 96: 217–248.

[2] Dicke, M., Takabayashi, J., Shutte, C.,Krips, O. E. Behavioural ecology of plant-carnivore interactions: variation in response of phytoseiid mites to herbivore-induced plant volatiles. *Experimental and Applied Acarology 22*: 595-601, 1997

[3] Fontana, W. and L.W. Buss, 1994. The arrival of the fittest: Toward a theory of biological organization. *Bulletin of Mathematical Biology* 56: 1–64. 1994.

[4] Maeda, T., Takabayashi, J.,Yano, S. and Takafuji, A. Factors affecting the resident time of the predatory mite Phytoseiulus persimilis (Acari: Phytoseiidae) in a prey patch *Applied Entomology and Zoology 33*: 573-576, 1998

[5] Păun, G. 1998. Computing with Membranes, *J. of Computer and System Sciences* (in press). (also on http://www.tucs.fi).

[6] Sabelis, M. W., and M.,C.M. de Jong, 1988. Should all plants recruit bodyguards? Conditions for a polymorphic ESS synomone production in plants *Oikos*, 53 247-252,

[7] Suzuki, Y. and H. Tanaka. 1997. Symbolic chemical system based on abstract rewriting system and its behavior pattern. *Journal of Artificial Life and Robotics*:1:211-219, Springer Verlag.

[8] Suzuki, Y. Tsumoto, S and H. Tanaka. 1996. Analysis of Cycles in Symbolic Chemical System based on Abstract Rewriting System on Multisets, *Artificial Life V*: 522-528. MIT press.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

An Approach to Cooperative Genetic Algorithms

Dongcheng Hu[1], Rui Jiang[1], Yupin Luo[1], K.Y.Szeto[2]

1 : Dept. of Automation, Tsinghua University, Beijing 100084, P.R.China
E-Mail : rjiang@mail.au.tsinghua.edu.cn

2 : Dept. of Physics, HKUST, Clear Water Bay, Hong Kong
E-Mail : phszeto@ust.hk

Abstract *: Based on the idea of combining the competition between agents with the cooperation between agents into a multi-agent computing system, this paper puts forward a cooperative genetic algorithm. The algorithm allocates the processor time that is originally exclusively occupied by a single simple genetic algorithm to several simple genetic algorithms that run concurrently on a serial computer and let these simple genetic algorithms exchange some individuals in each evolving cycle. A generic method for constructing such an algorithm is described. Main features of the cooperative genetic algorithm are analyzed. Experiments about function optimization problems and combinatorial optimization problems are designed and implemented to test the efficiency of the algorithm. Results show that our algorithm can improve the computing efficiency to a large extent.*

Key Words: *Genetic Algorithms, Cooperation, Competition, Multi-Agent Computing System*

1 Introduction

The genetic algorithm (GA) constitutes a large class of evolutionary computing methods with many applications in both technological and scientific research [1~6]. Based on the *Darwinian principle* of the survival of the fittest, genetic algorithms could deal with many challenging optimization problems that are difficult to solve with conventional techniques. Due to the ability of exploring different regions of the solution space in parallel and maintaining multiple solutions concurrently, genetic algorithms can execute search tasks quickly with relatively small and controllable probability to be trapped in local optima [1][2].

However, like other randomized algorithms such as the simulated annealing [7] and the ant system [8], the performance of the genetic algorithm varies greatly from one problem instance to another, even across repeated trials on a single problem instance [9]. The phenomenon of the unpredictable standard deviation in performance existing in the randomized searching algorithms can be characterized by a distribution describing the probability of obtaining each possible performance value. Although the mean or the expected values of these distributions are usually used as an overall measure of quality, the variance of a performance distribution also affects the quality of an algorithm to a great extent, since the variance of a performance distribution determines how likely it is that a particular run's performance will deviate from the expected one. The conception of the variance of a performance distribution is very similar to the conception of risk in economics. Indeed, risk is an important additional characteristic of algorithms because one may be willing to settle for a lower average performance in exchange for increased certainty in obtaining a reasonable answer. In analogy with the similar situation when trying to maximize a utility that has an associated risk encountered in the economics, Huberman and his collaborators have developed a method of combining existing algorithms into new programs that are unequivocally preferable to any of the component algorithms using the notion of risk in economics [9]. If we treat each component algorithm as an agent and treat the whole combined algorithm as a multi-agent system, we can see clearly that since the computing material, typically the processor time, is limited, agents in such a system must share the whole computing material and compete for these material. From this point of view, the main relationship between these agents is ***competition*** and it is the competition between these agents that helps the whole system to get the desirable performance.

On the other hand, the theoretical studies on the parallel genetic algorithms such as the study of the effects of dimensionality on parallel genetic algorithms by Dr. K.Y.Szeto has proved that the exchange of individuals between populations is one of the most important aspects for a parallel genetic algorithm [3~6] [10]. If we also treat each population in a parallel genetic algorithm as an agent and treat the whole parallel genetic algorithm as a multi-agent system, we can also see clearly that since agents in such a system exchange information with others, the main relationship between these agents is ***cooperation*** and it is the cooperation between these agents that helps the whole system to evolve toward a desirable direction.

Considering these, it is very natural to have the thought that if we combine the competition between

agents and the cooperation between agents into a multi-agent computational system, the performance of this system may reach an outstanding one. It is our aim in this paper to put forward a generic method to construct such a system and to analyze some main features of such a system.

The rest of this paper is organized as follows. In part 2, we will describe the method of constructing such a cooperative genetic algorithm. In part 3, we will describe some typical characteristic of the cooperative algorithm. In part 4, some experiments and their results will be discussed briefly. Finally, in part 5, we will give some conclusions and describe some of our future work.

2 Construct a cooperative genetic algorithm

The basic idea of constructing a cooperative genetic algorithm is to allocate the processor time that is originally exclusively occupied by a single simple genetic algorithm to several simple genetic algorithms that run concurrently on a serial computer and let these simple genetic algorithms exchange some individuals in each evolving cycle. The building block of our cooperative genetic algorithm is the simple genetic algorithm, which can be described as formula (1).

$$SGA = (P, S, C, M, U, T) \quad (1)$$

According to the formula, a simple genetic algorithm *SGA* consists of a population *P*, a selection method *S*, a crossover operator *C*, a mutation operator *M*, a survival strategy *U* and a termination criterion *T*. The evolving progress of such an *SGA* can be seen as a sequence of population P^i, (i=0,1,2,...). Here,

$$P^{i+1} = U(M(C(S(P^i)))) \quad (2)$$

Formula (2) means that the $(i+1)^{th}$ generation can be got by applying the selection method, the crossover operator, the mutation operator and the survival strategy one by one on the i^{th} generation. After the new generation has been created, the termination criterion is checked to determine whether the simple genetic algorithm should be stopped or not. If the criterion is satisfied, the algorithm is terminated; otherwise, the evolving progress should be continued.

For simplicity, we will assume that a cooperative genetic algorithm consists of two simple genetic algorithms with the same parameters, the same operators and the same strategies in the rest of this paper, although it is not a necessary requirement. Based on this assumption, a cooperative genetic algorithm can be described as formula (3).

$$CGA = (SGA_1, SGA_2, f, \tau, T) \quad (3)$$

According to the formula, a cooperative genetic algorithm consists of two simple genetic algorithms SGA_1 and SGA_2, a processor fraction f, an exchange fraction τ and a termination criterion *T*. The cooperative genetic algorithm is constructed simply by allocating the processor time that is originally exclusively occupied by a simple genetic algorithm to two simple genetic algorithms that run concurrently on a serial computer and letting these two algorithms keep contact with each other.

The whole evolution progress of this cooperative algorithm can be decomposed to several evolution steps. Each evolving step can be illustrated as Fig.1.

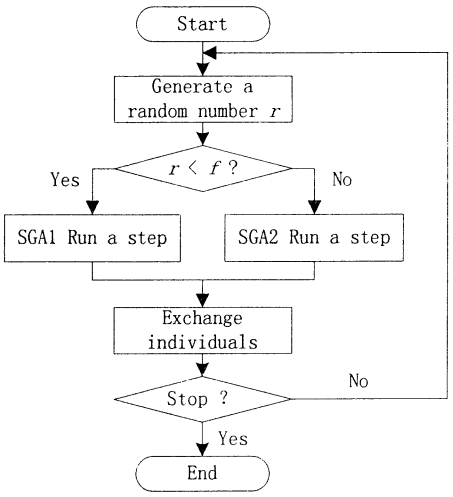


Fig.1 : A typical evolving step of the cooperative genetic algorithm

According to the figure, we can describe each evolving step of the algorithm as following sub-steps.

Step 1: Generate a random number $r \in [0,1]$;

Step 2: If $r < f$, let the SGA_1 evolve a generation, otherwise let the SGA_2 evolve a generation;

Step 3: Let the SGA_1 and the SGA_2 exchange some individuals, the number of these individuals is $\tau \times P$;

Step 4: Check the termination criterion. If the whole algorithm should be terminated, stop the algorithm; otherwise go to step 1.

We should also pay attention to several other things.

The exchange strategy: Each simple genetic algorithm uses the exchange strategy to select some individuals from its population and exchange them with the other simple genetic algorithm. The simplest strategy is random selection. There are also some complex strategies such as random selection plus the best ones, random selection plus the worst ones, and so on.

The termination criterion: The whole algorithm use the termination criterion to stop the running of the algorithm. The simplest criterion is to check the termination criteria of both component genetic algorithms and use their logical or result. That is, if any component genetic algorithm should be terminated, the whole cooperative algorithm should also be terminated.

The symmetry: There are some symmetric features existing in our algorithm. Since we assume that each component simple genetic algorithm has same parameters, we can get that

$$CGA(SGA_1, SGA_2, f, \tau, T) = CGA(SGA_1, SGA_2, 1-f, \tau, T)$$

and

$$CGA(SGA_1, SGA_2, f, 1, T) = CGA(SGA_1, SGA_2, 0.5, 0, T)$$

Here, we assume that the total process time is 1.

3 Main features of the cooperative genetic algorithm

In this part, we will focus on two control parameters — the processor fraction (f) and the exchange fraction (τ) and two performance variables — the mean or expected value (m) and the standard deviation (σ) of the performance distribution. Fig.2 shows the typical relationship of the mean value versus the processor fraction and the exchange fraction. The figure describes the typical relationship of the standard deviation value versus the processor fraction and the exchange fraction is very similar to Fig.2, so we omit that figure here.

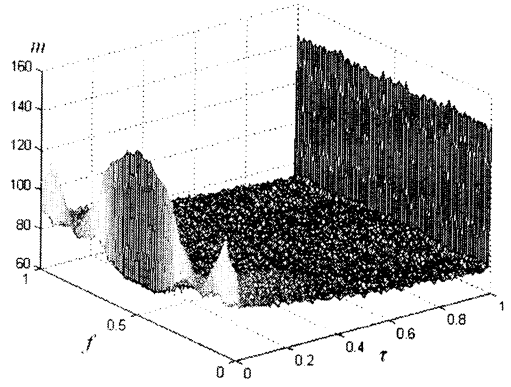


Fig.2 : The typical relationship of the mean value versus the processor fraction and the exchange fraction.

By keeping the exchange fraction τ as a constant, we can get the curve of the mean value versus the processor fraction and the curve of the standard deviation value versus processor time, both of which are shown in Fig.3

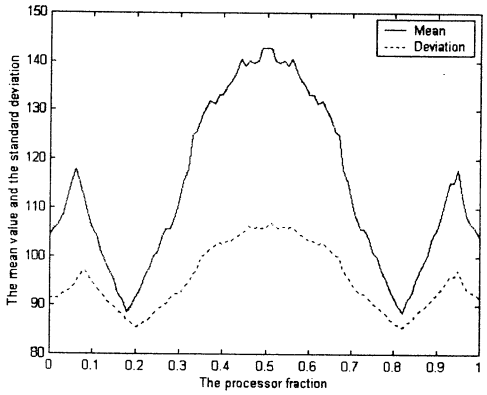


Fig.3 : The curve of the mean value and the standard deviation versus the processor fraction

We can even get the curve of the mean versus the standard deviation value, which is shown in Fig.4.

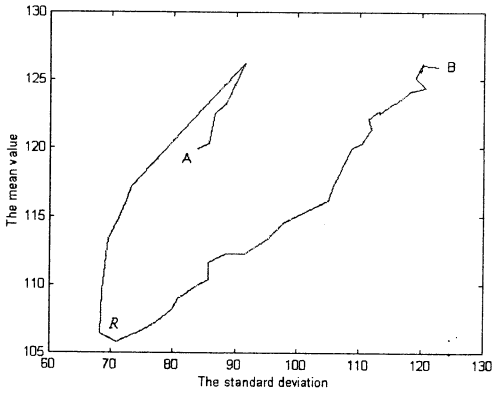


Fig.4 : The curve of the mean value versus the standard deviation

The curve shown in Fig.4 is very similar to the ***efficient frontier*** curve discussed by Huberman in [9]. There are several features of the curve that are worth noting. First, endpoint A corresponds to the situation that there is just one simple genetic algorithm running alone (f = 0 or 1), and endpoint B corresponds to the situation that both simple genetic algorithms sharing computer time equally (f = 0.5). Second, there exists a regime, the efficient frontier R, defined by the fact that for every point on the curve, there is at least one point on the efficient frontier that is always preferable, that is, has a lower standard deviation value or higher mean value, or both. Once this efficient frontier is determined, one can choose the desired mean-deviation combination on it and calculate the corresponding fraction of computer cycles to be allocated to algorithm 1. This calculation can be done by plotting both the mean value and the standard deviation value as a function of f (As is shown in fig.3).

If we keep the processor fraction f as a constant, we can get the curve of the mean value versus the exchange fraction and the curve of the standard deviation value versus the exchange fraction, both of which are shown in Fig.5

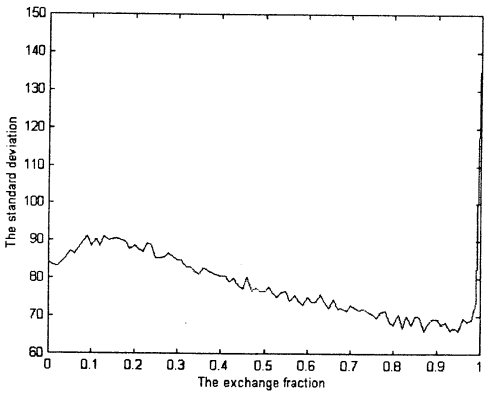


Fig.5 : The curve of the standard deviation value versus the exchange fraction

There are also several features of the curve shown in Fig.5 that are worth noting. First, we can see from the figure that the exchange fraction or the information exchange level has large effect on the performance of the algorithm, since the performance with some information exchange is much better than the one without any information exchange. Second, if the information exchange level is too high, the performance will drop either. This phenomenon is not surprising since too much information exchange may destroy the internal information chain existing in the simple genetic algorithm.

4 Experiments and Results

In order to test the efficiency of our cooperative genetic algorithm, we have used our generic method to construct corresponding algorithms for function optimization problems and combinatorial optimization problems. As for the function optimization problems, we use the simple genetic algorithm and the cooperative genetic algorithm to solve the standard DeJong functions [1][2] and compare the mean and the standard deviation values of their distributions. As for the combinatorial optimization problems, we use the simple genetic algorithm and the cooperative genetic algorithm based on it to solve the standard traveling salesman problem and compare the mean and the standard deviation values of their distributions. Results show that with proper parameters, our cooperative algorithm can improve the computational efficiency to a large extent. We will discuss the result in a more detail way in another paper.

5 Conclusions

In this paper, we combined the idea of competition between agents with the idea of cooperation between agents into a multi-agent computing system and put forward a cooperative genetic algorithm. After describing the generic method of constructing such a cooperative genetic algorithm, we discussed the main features of such an algorithm. In order to test the efficiency of our algorithm, we have constructed and implemented corresponding algorithms for function optimization problems and combinatorial optimization problems. Experiments results show that our method can improve the computation efficiency to a large extent.

There are also some other things that are worth noting in our future work. First, the competition is now limited to the processor time allocating, but we may develop this idea and construct some more complicated algorithms such as some multiple-level cooperative genetic algorithms. Second, we may incorporate the idea of competition and the idea of cooperation into some other randomized searching algorithms such as the ant system. By doing this, we may construct more algorithms that are more efficient in their domains. Third, although our algorithm now runs only on a single serial computer, it is potential to be extended to more complex computing environment such as a parallel computer. Finally, although we do not consider the distributed computing environment in our current algorithm, but if we consider the existing asymmetry in the distributed computing environment, we may develop our idea and method and extend them to the distributed computing environment.

References

[1] J.H. Holland, Adaptation in Natural and Artificial Systems, Ann Arbor, *MI: University of Michigan Press*, 1975.

[2] Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning. *Addison-Wesley, Reading, MA*, (1989)

[3] S.P. Li and K.Y. Szeto; Crytoarithmetic problem using parallel Genetic Algorithms.(Mendl'99, Brno, Czech, 1999, June).

[4] K.Y. Szeto and K.H. Cheung; Multiple Time Series Prediction using Genetic Algorithms Optimizer, Proceedings of the International Symposium on Intelligent Data Engineering and Learning, Hong Kong, IDEAL'98, p.127-133, 1998.

[5] K.Y. Szeto and K.H. Cheung; Annealed genetic algorithm for multiple time series prediction, Proceedings of the World Multiconference on Systemic, Cybernetics and Informatics, Caracas, V.3, 390-396(1997).

[6] K.Y. Szeto and P.X. Luo; Self Organized Genetic Algorithm in forecasting stock market.(invited talk at the FFM'99, the Proceeding of the Sixth International Conference 'Forecasting Financial Markets' , London, 26-28 May 1999, CD-ROM)

[7] S.Kirkpatrick, C.D.Gelatt, Jr., M.P. Vecchi, Optimization by Simulated Annealing. *Science*, 13 May 1983, Volume 220, Number 4598

[8] Marco Dorigo, Vittorio Maniezzo, and Alberto Colorni, Ant System : Optimization by a Colony of cooperating Agents. IEEE Transactions on System, Man, and Cybernetics-Part B:Cybernetics, Vol, 26, No.1, pp 29-41, 1996.

[9] B.A.Huberman, R.M.Lukose, and T.,Hogg, An economics approach to hard computational problems, Science, Vol.275, 51-54, 1997.

[10] K.Y. Szeto , K.H. Cheung and S.P. Li; Effects of Dimensionality in Parallel Genetic Algorithms, Proceedings of the World Multiconference on Systemic, Cybernetics and Informatics, Orlando, ISAS'98, Ed. by N.Callaos, T.Yang, and J. Aguilar, Vol.2, 322-326,(1998).

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Optimal Routing and Flow Control for Multiple I/O Data Network by using Genetic Algorithm

K. OKUHARA†, T. OKADA† and T. TANAKA†

† Dept. of Management and Information Sciences,
Hiroshima Prefecture University,
562, Nanatsuka, Shyobara 727-0023, Japan.

Abstract

In this paper we propose an application of a genetic algorithm to control the information network. The communication channel and the communication control processor are finite resource. We must use these resources efficiently to improve the quality of service (QoS). For this purpose we apply an evolution computation such as a genetic algorithm to an optimal routing and flow control.

1 Introduction

An optimal routing and a flow control are important problems in multiple I/O data network. Recently, the computer network such as an Internet which was developed from the ARPAnet has come to be a huge system. Therefor it becomes difficult to solve the routing problem and the flow control problem, because these problems are formulated as a constrained nonlinear optimization problem which has many local minimum points.

The genetic algorithm (GA) is one of powerful tool for searching a global minimum point of the constrained nonlinear optimization problem[1]. It is known that the GA is available to solve the routing problem such as the travelling salesman problem. However there are few studies which use the GA for solving the routing problem and the flow control problem simultaneously. The Dijkstra's algorithm[2] has been applied to the routing.

The purpose of this paper is to apply the GA to solve both the routing problem and the flow control problem. The proposed method can realize an adaptive routing and a distributed flow control by using a flooding algorithm which executes a broadcast of flow information.

Simulation results say that our method can not only search the optimal routing among Origin - Destination pair but also control the flow of link. Furthermore compared with the usual method named a flow deviation method, we show that the proposed method can find a minimum solution.

2 Multiple I/O Data Network

We assume that a network consists of N nodes. The packet switching method is used as the method of communication. Assuming that a packet switching equipment is placed in each node. Data is sent by links among nodes. Node i, $(i = 1, 2, \cdots, N)$ is connected to neighboring node j by a link L_{ij}, $(j \in L_i)$, where L_i denotes a connection from node i and a link capacity of L_{ij} is assumed to be finite. The conceptual figure of information network is shown by figure 1. Assuming that there is infinite waiting buffer for each link in node i. The service rule in the waiting buffer is the first in first out (FIFO). We furthermore assume that there are a topology table which describes the network topology information and a traffic table which describes the link traffic information at all nodes.

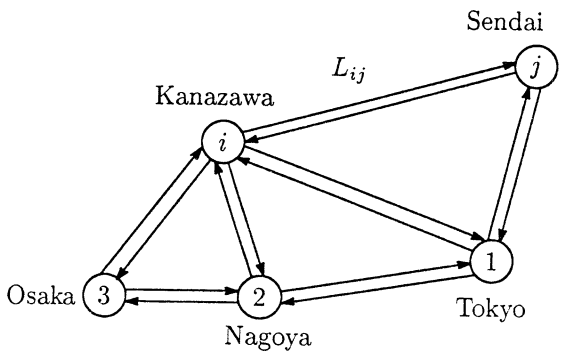


Figure 1 Conceptual figure of information network.

In this study, the network routing is executed by a virtal circuit. The topology and traffic tables are

updated by the flooding when the network detects its change. The session w which has an origin o and a destination d is represented by the pair of them (OD pair) such as $w = (o, d)$. W denotes a set of OD pair, and P_w all subsets of the directed path which connects the origin and the destination of session w. Assuming that the arrival rate (flow) of an input traffic relating to a path p of a session w is described by $x_{w,p}$[pkt/s], the total flow $f_{ij}(\mathbf{x})$[pkt/s] in the link L_{ij} is given by

$$f_{ij}(\mathbf{x}) = \sum_{p \in p_{ij}} x_{w,p} \quad (w \in W)$$

where $\mathbf{x}$ denotes a flow vector which consists of all flow $x_{w,p}$ and p_{ij} a set of paths which include the link L_{ij}. And the total flow of an input traffic at node o in session w is described by r_w[pkt/s].

Suppose that the network can detect the link capacity c_{ij}[pkt/s] at all link L_{ij} and the delay d_{ij}[s/pkt] at output links from node. In the flow model base on the M/M/1 approximations, the flow vector $\mathbf{x}$ which minimize an average of packet number in the network is obtained by the solution of the following nonlinear optimization problem (NLP)[3].

Minimize

$$D(\mathbf{x}) = \sum_{i=0}^{N}\sum_{j=0}^{N}\left(\frac{f_{ij}(\mathbf{x})}{c_{ij} - f_{ij}(\mathbf{x})} + d_{ij} f_{ij}(\mathbf{x})\right) \equiv \sum_{i=0}^{N}\sum_{j=0}^{N} D_{ij}\{f_{ij}(\mathbf{x})\}$$

Subject to

$$\sum_{p \in P_w} x_{w,p} = r_w \quad (w \in W),$$
$$0 \le f_{ij}(\mathbf{x}) \le c_{ij} \quad (w \in W,\ p \in P_w).$$

3 Routing and Flow Control by Usual Method

3. 1 Routing by Dijkstra's Algorithm

The first derivative function of the average packet number $D(\mathbf{x})$ with respect to the flow $x_{w,p}$ is given by

$$\frac{\partial D(\mathbf{x})}{\partial x_{w,p}} = \sum_{L_{ij} \in p}\left[\frac{c_{ij}}{\{c_{ij} - f_{ij}(\mathbf{x})\}^2} + d_{ij}\right] \equiv \sum_{L_{ij} \in p} D'_{ij}\{f_{ij}(\mathbf{x})\}.$$

It can be considered as the length of the path p, that is, this is called the first derivative length (FDL) of the path p. The path which minimizes the FDL is called a MFDL. In usual routing of the flow model, the shortest path problem has been solved by the Dijkstra's algorithm.

The second derivative function of the average packet number $D(\mathbf{x})$ with respect to the flow $x_{w,p}$ is also given by

$$\frac{\partial^2 D(\mathbf{x})}{(\partial x_{w,p})^2} = \sum_{L_{ij} \in p} \frac{1}{\{c_{ij} - f_{ij}(\mathbf{x})\}^3} \equiv \sum_{L_{ij} \in p} D''_{ij}\{f_{ij}(\mathbf{x})\}.$$

3. 2 Flow Control by Gradient Method

We introduce a constant $\rho < 1$ to ease restrictions on the total flow $f_{ij}(\mathbf{x})$ of the link L_{ij} in NLP. Then we consider a function $\tilde{D}_{ij}\{f_{ij}(\mathbf{x})\}$ which has the same value of the first and second derivative of the function $D_{ij}\{f_{ij}(\mathbf{x})\}$ at $f_{ij}(\mathbf{x}) = \rho c_{ij}$. Figure 2 shows the difference among these functions.

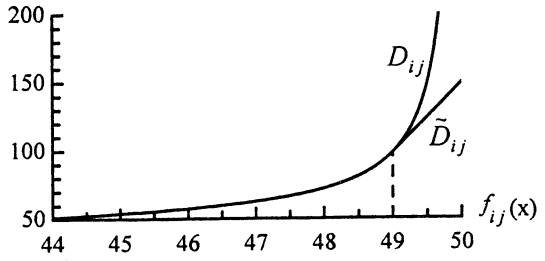


Figure 2 Difference between the function $D_{ij}\{f_{ij}(\mathbf{x})\}$ and $\tilde{D}_{ij}\{f_{ij}(\mathbf{x})\}$. ($c_{ij} = 50$ [pkt/s], $d_{ij} = 1$ [s/pkt] and $\rho = 0.98$)

The function $\tilde{D}_{ij}\{f_{ij}(\mathbf{x})\}$ is given by

$$\tilde{D}_{ij}\{f_{ij}(\mathbf{x})\} = \begin{cases} D_{ij}\{f_{ij}(\mathbf{x})\}, & (f_{ij}(\mathbf{x}) \le \rho c_{ij}) \\ \alpha_{ij}(f_{ij}(\mathbf{x}) - \rho c_{ij})(f_{ij}(\mathbf{x}) - \beta_{ij}) & \\ \quad + D_{ij}\{\rho c_{ij}\}, & (f_{ij}(\mathbf{x}) > \rho c_{ij}) \end{cases}$$

where

$$\alpha_{ij} = \frac{1}{2} D''_{ij}\{\rho c_{ij}\},$$
$$\beta_{ij} = \rho c_{ij} - \frac{1}{\alpha_{ij}} D'_{ij}\{\rho c_{ij}\}.$$

We further describe the MFDL of each OD pair w by $\bar{p}$ and give the flow $x_{w,\bar{p}}$ such as

$$x_{w,\bar{p}} = r_w - \sum_{p \in P_w,\ p \neq \bar{p}} x_{w,p}.$$

Thus the NLP can be translated into the following nonlinear optimization problem (NLP');

Minimize

$$\tilde{D}(\tilde{\mathbf{x}}) = \sum_{i=0}^{N} \sum_{j=0}^{N} \tilde{D}_{ij}\{f_{ij}(\tilde{\mathbf{x}})\}$$

Subject to

$$0 \leq x_{w,p} \quad (w \in W,\ p \in P_w,\ p \neq \bar{p})$$

where $\tilde{\mathbf{x}}$ consists of all flow vector which is excepting the MFDL.

The first derivative function of the average packet number $\tilde{D}(\tilde{\mathbf{x}})$ with respect to the flow $x_{w,p}$ is given by

$$\frac{\partial \tilde{D}(\tilde{\mathbf{x}})}{\partial x_{w,p}} = \frac{\partial \tilde{D}(\mathbf{x})}{\partial x_{w,p}} - \frac{\partial \tilde{D}(\mathbf{x})}{\partial x_{w,\bar{p}}}, \quad (w \in W,\ p \in P_w,\ p \neq \bar{p})$$

where the first term of the right hand is defined by

$$G_{w,p} = \frac{\partial \tilde{D}(\mathbf{x})}{\partial x_{w,p}} \equiv \sum_{L_{ij} \in p} \tilde{D}'_{ij}\{f_{ij}(\mathbf{x})\}.$$

Then the function $\tilde{D}'_{ij}\{f_{ij}(\mathbf{x})\}$ is a modified MFDL and it is derived by

$$\tilde{D}'_{ij}\{f_{ij}(\mathbf{x})\} = \begin{cases} D'_{ij}\{f_{ij}(\mathbf{x})\}, & (f_{ij}(\mathbf{x}) \leq \rho c_{ij}) \\ \alpha_{ij}(2f_{ij}(\mathbf{x}) - \beta_{ij} - \rho c_{ij}), & (f_{ij}(\mathbf{x}) > \rho c_{ij}) \end{cases}$$

The second derivative function of the average packet number $\tilde{D}(\tilde{\mathbf{x}})$ with respect to the flow $x_{w,p}$ is given by

$$H_{w,p} = \frac{\partial^2 \tilde{D}(\tilde{\mathbf{x}})}{(\partial x_{w,p})^2} \equiv \sum_{L_{ij} \in L_p} \tilde{D}''_{ij}\{f_{ij}(\tilde{\mathbf{x}})\}$$

where

$$\tilde{D}''_{ij}\{f_{ij}(\tilde{\mathbf{x}})\} = \begin{cases} D''_{ij}\{f_{ij}(\tilde{\mathbf{x}})\}, & (f_{ij}(\tilde{\mathbf{x}}) \leq \rho c_{ij}) \\ 2\alpha_{ij}, & (f_{ij}(\tilde{\mathbf{x}}) > \rho c_{ij}) \end{cases}$$

L_p denotes the set of link which belong to either the path p or the MFDL($\bar{p}$).

In the usual flow model, the flow control has been executed by the following gradient method

$$x_{w,p}^{k+1} = \max\left[0,\ x_{w,p}^{k} - \epsilon^k H_{w,p}^{-1}(G_{w,p} - G_{w,\bar{p}})\right].$$

4 Routing and Flow Control by GA

4. 1 Routing by GA

Supposing that there is the topology table which describes the topology information of network at all nodes. The content of topology table is a neighbor node j which is connected from node i and it is updated when the network detects changes. The chromosome consists of H, $(\leq N)$ genes, and h, $(h = 1, \cdots, H)$th locus denotes hth trough point of the virtual path. Table 1 shows an example of topology table.

Table 1 Example of topology table

	Connection Node				
Node	1	2	3	4	5
1	2	4	0	0	0
2	1	3	4	5	6
⋮	⋮	⋮	⋮	⋮	⋮
9	5	6	8	0	0

Figure 3 shows an example of chromosome ($H = N$) which is generated for the OD pair $(1, 9)$, where the lethal chromosome is marked by $\times$.

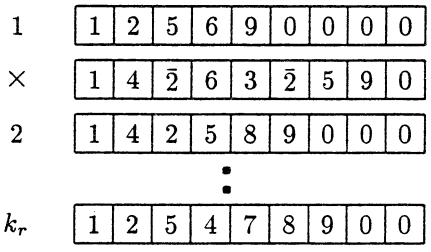


Figure 3 Example of initial population. for OD pair (1,9)

In the routing by GA, the reproduction is executed by the roulette selection. The crossover is executed by the one-point crossover. The mutation is not applied to the routing.

4. 2 Flow Control by GA

In the flow control by GA, the crossover is executed by the whole arithmetical crossover. The mutation

is executed by both uniform mutation and boundary mutation.

Figure 4 shows the structure of chromosome for the flow control by GA. An example of the initial population in feasible region is shown by figure 5.

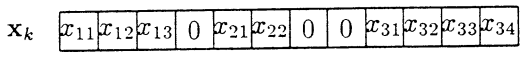


Figure 4 Structure of chromosome for flow control.

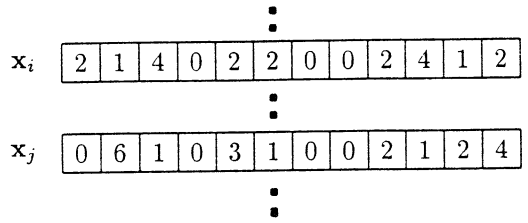


Figure 5 Initial population in feasible region.

5 Simulation Results

We consider the lattice type network (Figure 3).

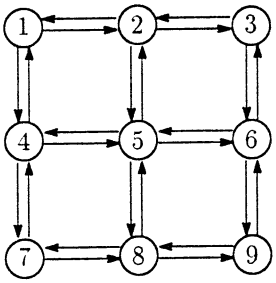


Figure 6 Structure of lattice type network (3^2).

In order to compare the validity by difference of the network size, we first assume that sessions occur in same rate for the node number of network. Therefor we assume that the network has N nodes and $1/\sqrt{N}$ sessions. Figure 7 shows the comparison of validity by difference of the network size.

In order to compare the validity by difference in ratio of session, we secondly assume that sessions occur in different rate for the constant node number of network. Figure 8 shows the comparison of validity by difference in ratio of session.

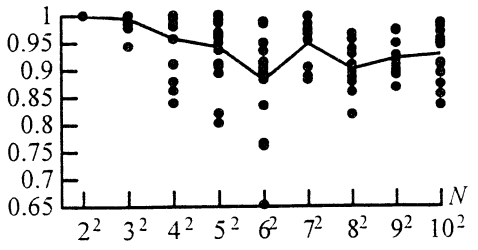


Figure 7 Comparison of validity by difference of the network size.

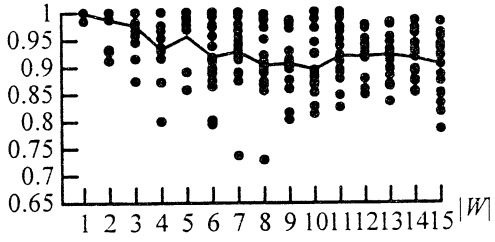


Figure 8 Comparison of validity by difference in ratio of session.

6 Summary

In this study we proposed the routing and the flow control by applying genetic algorithm to the flow model which can evaluate the congestion of network by the arrival rate of traffic.

From the results of simulation, compared with usual method, it was shown that the proposed method can reduce the average of packet number in networks. And we can say that proposed method has the effect which makes the ratio used of link uniform.

References

[1] J. H. Holland, "*Adapptation in Natural and Artifical Systems*," University of Michigan Press, 1975.

[2] E. W. Dijkstra, "A Note of Two Problems in Connection with Graphs," *Numerische Mathematik*, **1**, pp. 269-271, 1959.

[3] D. Bertsekas and R. Gallager, "Data Networks," *Prentice-Hall, Inc.*, 1987.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Emergence of Cognitive Robot Behavior Using Affordance and Genetic Algorithm

Kiyoharu Tagawa†, Katsumi Inoue and Hiromasa Haneda
†Dept. of Electrical and Electronics Engineering, Kobe University
Rokkodai-Cho 1-1, Nada-Ku, Kobe City 657-8501 Japan
E-mail: tagawa@eedept.kobe-u.ac.jp

Abstract

The objective of this paper is to demonstrate the possibility of behavior-based intelligence from a viewpoint of the affordance theory. A virtual robot that executes various tasks demanded in several different environments is described. Through the interaction with surrounding environment, the robot recognizes not only current environment but also its own posture successfully, whereas the robot has no map representing the world in which it is going to act. For the great work, the robot finds and uses appropriate affordances lurking in the surrounding environment.

1 Introduction

The essential ingredients of vital intelligence depend on the kinds of experience that comes from the interaction between living bodies and their surrounding environments. In order to formulate and examine such a new Artificial Intelligence (AI) paradigm, robotic agents have been studied intensively[1].

R. A. Brooks[2] showed the possibility of behavior-based intelligence by making a behavior-based robot known as subsumption architecture[3]. After that, the behavior-based approach becomes a dominant methodology for realizing the new AI paradigm[4, 5]. Behavior-based robots usually have no map that represents their world, however they can take appropriate motion through the interaction with their surrounding environments. Although many successful robots have been built up based on this approach, until recently, little attention has been paid to the interaction between robots and their environments.

In this paper, instead of the mechanical structure of behavior-based robots, we focus our attention on the interaction between robots and environments based on the affordance theory. The "affordance" advocated by J. J. Gibson[6] is an important concept in the research fields of perception and cognition. The affordance exists in their surrounding environments and affords the potentialities of motions to living bodies.

We present a virtual robot that behaves itself like an intelligent one by detecting useful affordances in its surrounding environment. Through the active interaction with environments, the robot not only distinguishes between environments, but also recognizes its own posture successfully, whereas the robot has no map representing the world in which it is going to act. The architecture of the robot is separated into two parts, namely, brain and body. The robot's brain is defined by a Finite State Machine (FSM), which is evolved by using a Genetic Algorithm (GA). On the other hand, the robot's body moves according to the deterministic action command given by FSM. Tracing the behavior of the robot, we specify two types of affordance in each environment. The positive affordance allows the robot to take intentional actions, while negative affordance forbids the robot to take them. Then, analyzing the contribution of these affordances to the robot's motion, we try to explain the origin of the behavior-based intelligence.

2 Robot and Environment

Our image of an intelligent robot is illustrated in Fig.1. We suppose that there are several different environments in which the robot has to play an active part. The robot's body happens to be surrounded with one of the environments. Then, the robot is requested to distinguish the current environment from the others by itself and execute the task demanded in each environment as soon as possible.

The process of robot's brain is described as a Finite State Machine (FSM), which is defined by a sequence $(Q,\ C,\ I,\ \delta_c,\ \delta_t,\ q_0)$: a set of state Q, a set of action command C, a set of input $I = \{0,\ 1\}$, a command function $\delta_c : Q \rightarrow C$, a transition function $\delta_t : Q \times$

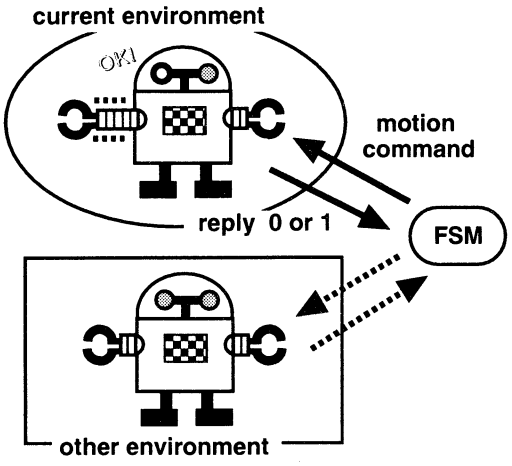


Figure 1: Image of robot and environment

$I \rightarrow Q$, and an initial state $q_0 \in Q$. First of all, the command function $\delta_c : Q \rightarrow C$ is evaluated at the initial state $q_0 \in Q$ of FSM. The action command $c_k \in C$ associated with the state orders the robot's body one single simple motion, probably of limited extent, such as move jaws right one unit length. If possible, the robot's body takes the action and returns symbol $1 \in I$ to the FSM; otherwise it does not move and returns symbol $0 \in I$. Then, according to the robot's reply, the transition function $\delta_t : Q \times I \rightarrow Q$ is evaluated to obtain the next state.

3 Genetic Algorithm

In order to design an intelligent robot's brain, i.e., an appropriate structure of FSM, we employ a GA. The plural of FSM is created randomly and modified repeatedly by using genetic operations.

3.1 Lamarckian Mutation

In the mutation of FSM, a state $q_j \in Q$ is selected and modified by one of the following operations: 1) Replace the action command $\delta_c(q_j) \in C$; 2) Change the value of the transition function $\delta_t(q_j, 0)$ and $\delta_t(q_j, 1)$; 3) Duplicate the state $q_j \in Q$; 4) Delete the state $q_j \in Q$. The above four operations have been also adopted by the Evolutionary Programming[8].

In the Lamarckian mutation of FSM, which has been proposed in our previous paper[7], we chose a target state $q_j \in Q$ by using the probability proportional to its weight $W(q_j)$ defined as follows.

$$\begin{cases} W(q_j) = BASE + TAX * S_1(q_j) \\ \qquad\qquad + FINE * S_0(q_j) \\ W(q_j) = 0, \quad (\text{iff } S_0(q_j) = S_1(q_j) = 0) \end{cases} \tag{1}$$

where, $BASE$, TAX and $FINE$ are positive scaling parameters; $S_0(q_j)$ and $S_1(q_j)$ denote the total numbers of 0 and 1 replied to $q_j \in Q$ respectively.

3.2 Multiple Fitness Function

We suppose that there are N different environments E_n $(n = 1 \sim N)$ in which a virtual robot has to execute its task. The computational time of FSM is evaluated by the total number of the state transition executed until the robot completes the task in each environment. Therefore, we measure the performance of FSM by the computational time and define a fitness function of FSM in each environment E_n as,

$$fitness(FSM, E_n) = \frac{\alpha_n}{time(FSM, E_n)} \tag{2}$$

where, $time(FSM, E_n)$ denotes the computational time of FSM spent in the environment E_n; α_n is a coefficient for scaling the value of the fitness.

Then, we define a multiple fitness function as,

$$fitness(FSM) = \sum_{n=1}^{N} \frac{fitness(FSM, E_n)}{N}. \tag{3}$$

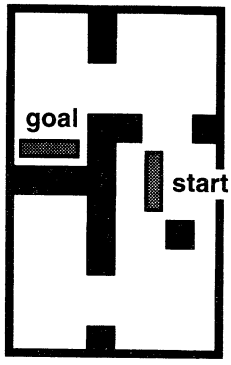
3.3 Generation Alternation Model

Since we have to adapt the robot's brain (FSM) to several different environments E_n $(n = 1 \sim N)$, we use the techniques of applying GAs to multi-criteria optimization problems[9].

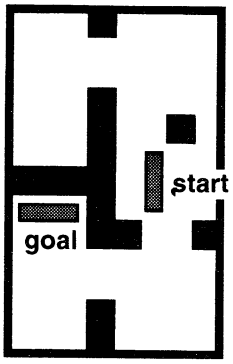
In the genetic alternation model, the selection of parental FSMs is carried out in parallel by the roulette wheel algorithm in the light of their fitness in (2) for each environment E_n. At the same time, some FSMs are also selected from the current population according to their multiple fitness in (3). Then, in order to create the next generation, the Lamarckian mutation and one-point crossover are applied to them. Elitist selection is also employed in parallel, namely the best FSM always survives to the next generation.

4 Experimental Results

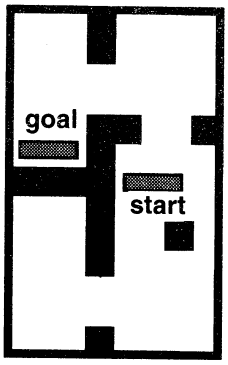
We have conducted experiments on an extended version of the well-known "carrying sofa problem"[10].



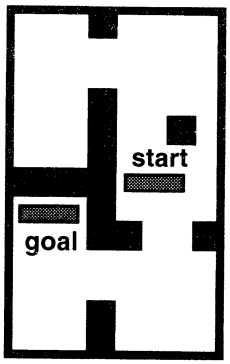
a) Environment E_1 (room-A)



b) Environment E_2 (room-B)



c) Environment E_3 (room-A)

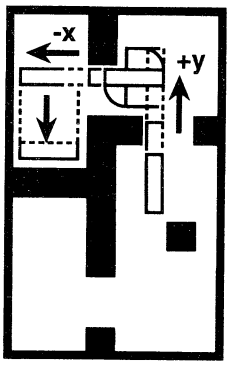


d) Environment E_4 (room-B)

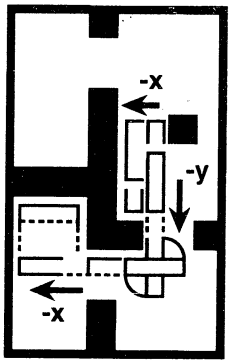
Figure 2: Four environments of robot

Q	δ_c	δ_t		Q	δ_c	δ_t	
		0	1			0	1
q_0	$+y$	q_2	q_1	q_6	$-y$	q_{10}	q_7
q_1	$-x$	q_0	q_2	q_7	$-y$	q_{12}	q_2
q_2	R	q_3	q_1	q_8	$+y$	q_{11}	q_9
q_3	$-x$	q_5	q_4	q_9	$+y$	q_{11}	q_7
q_4	$-x$	q_7	q_{13}	q_{10}	$+x$	q_6	q_5
q_5	$-y$	q_1	q_6	q_{13}	$-x$	q_5	q_8

Figure 3: Tabular form of the best FSM



a) Trajectory in E_1 (room-A)



b) Trajectory in E_2 (room-B)

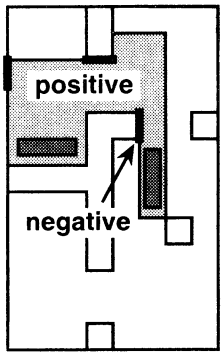
Figure 4: Trajectories of robot's body

As you can see in Fig.2, there are two symmetrical rooms, room-A and room-B, in which the robot has to move from start to goal avoiding obstacles as soon as possible. Painted area denotes wall and obstacles. The robot must distinguish the current room from the other by itself. Furthermore, we suppose that the robot has a rectangular body and takes two different starting postures in each room. Consequently, there are four different environments $E_1 \sim E_4$ for the robot. Five action commands are allowed to the robot: 1), 2) move $\pm x$ direction one unit length; 3), 4) move $\pm y$ direction one unit length; 5) rotate $R = 90°$.

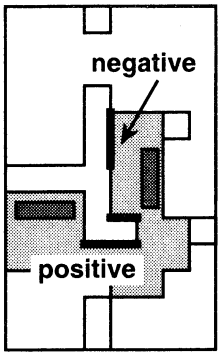
Figure 3 shows a tabular form of the best FSM (only valid states) genetically evolved. The table lists the states $q_j \in Q$ coupled with an action command $\delta_c(q_j)$ down the side and inputs 0, 1 $\in I$ across the top. The column under each input gives the value of the transition function $\delta_t : Q \times I \rightarrow Q$ corresponding to that input at each state shown on the left side.

The virtual robot loaded with the best FSM moves from start to goal in 18, 17, 21 and 20 unit times, respectively, within the four environments $E_1 \sim E_4$ shown in Fig.2. Figure 4 illustrates different trajectories of the robot's body starting from the same initial posture in room-A and room-B. From the results in Fig.4, we can say that the robot is able to distinguish two rooms by itself, because it has arrived at respective goals directly. Note that we can not intuitively predict such an intelligent performance of the robot from the simple structure of the FSM in Fig.3. This means that the cognitive behavior emerges from the interaction between robot and environments.

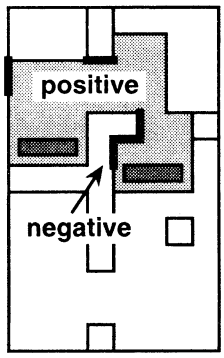
Figure 5 indicates two types of affordances in each environment, i.e., a part of environments concerned in the robot's behavior. The positive affordance light-shaded permits the robot's actions ordered by the FSM in Fig.3. On the other hand, the negative affordance dark-shaded prohibits them. We can ignore all other parts of environments, because they do not affect robot's motion. By tracing the behavior of the robot within the respective environments, we could find that the robot recognizes not only current room but also its own starting posture by using the positive and the negative affordances.



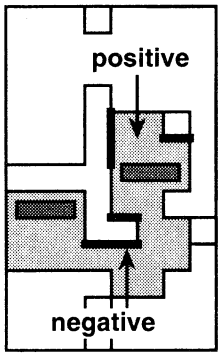
a) Affordances in E_1 (room-A)



b) Affordances in E_2 (room-B)



c) Affordances in E_3 (room-A)



d) Affordances in E_4 (room-B)

Figure 5: Affordances found and used by robot

5 Conclusions

In this paper, we have presented a virtual robot that recognizes not only current environment but also its own posture through the interaction with surrounding environment. From a view-point of the affordance theory, we have traced the amazing robot's behavior to the origin of its intelligence. As a result, we have specified positive and negative affordances characterized by robot's actions in each environment.

The results and analysis reported here are only the beginning of our continuing research project. Further work will consider the affordance in the world for a real robot with physical body. In order to depict useful affordances within more complex environments, an effective measure will have to be developed.

Acknowledgments

We wish to thank the Hyogo Science and Technology Association for their generous financial assistance. Most of the work on the implementation of GA program has been carried out by Shunsuke Kawaguchi.

References

[1] L. Steels and R. Brooks Eds., *The Artificial Life Route to Artificial Intelligence: Building Embodied Situated Agents.* Lawrence Erlbaum Associates, 1995.

[2] R. A. Brooks, "Intelligence without representation," *Artificial Intelligence*, vol.47, pp.139–159, Elsevier Science, 1991.

[3] R. A. Brooks, "A robust layered control system for a mobile robot," *IEEE Journal of Robotics and Automation*, vol.RA-2, no.1, pp.14–23, 1986.

[4] M. Colombetti, M. Dorigo and G. Borghi, "Behavior analysis and training - a methodology for behavior engineering," *IEEE Trans. on Systems, Man, and Cybernetics - Part B: Cybernetics*, vol.26, no.3, pp.365–380, 1996.

[5] M. Sugisaka, "Design of an artificial brain for robots," *Proc. of 3rd Int. Symp. on Artificial Life, and Robotics*, pp.2–11, 1998.

[6] J. J. Gibson, *The Ecological Approach to Visual Perception.* Houghton Mifflin Company, 1979.

[7] K. Tagawa, S. Kawaguchi, H. Iwamoto, K. Inoue and H. Haneda, "Automatic programming of autonomous robots using a genetic algorithm," *Proc. of 3rd Int. Symp. on Artificial Life, and Robotics*, pp.289–292, 1998.

[8] D. B. Fogel, "An introduction to simulated evolutionary optimization," *IEEE Trans. on Neural Networks*, vol.5, no.1, pp.3–14, 1994

[9] C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multi-objective optimization: formulation, discussion and generalization," *Proc. of 5th Int. Conf. Genetic Algorithms*, pp.303–310, 1993.

[10] D. E. Whitney, "State space models of remote manipulation tasks," *IEEE Trans. on Automatic Control*, vol.AC-14, no.6, pp.617–623, 1969.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15–17, January, 2001,

A Hierarchical Parallel Distributed Genetic Algorithm

T.Matsumura, M.Nakamura, S.Tamaki
University of the Ryukyus
Department of Information Engineering
1 Senbaru, 903–0213 Japan

K.Onaga
Okinawa Research Center
TAO
1 Asahimachi, 900–0029 Japan

Abstract

In this paper, we propose the hierarchical PDGA and evaluate its effectiveness compared Cone with lattice topology. We find that our proposed hierarchical PDGA obtains almost same or better solution than Cone. Moreover, we propose and investigate the processing time by the hierarchical PDGA on bus connection.

key words: hierarchical PDGA, topology

1 Introduction

Genetic algorithms (abbreviated to GAs) [1, 2] are remarkable one of meta-heuristics to solve combinatorial optimization problems. Many papers report its effectiveness and usefulness [1], but GAs generally take expensive computation cost. As a straightforward approach to overcome the problem, many proposals on parallel GAs are appeared [3, 4, 5, 6, 7, 8]. The distributed computation of parallel GAs contributes to keeping variety of chromosomes to avoid immature convergence in which the chromosome set is divided into several groups and is operated independently [5, 6].

A Parallel and Distributed GA (PDGA) has been proposed in [9] where we showed its effectiveness and influence of the network topology of the parallel machine. In that paper, *Cone* topology obtained better solution quality than the others, *ring*, *torus*, and *hypercube*. Cone topology is regarded as hierarchical connection of ring topologies. Each layer of the topology corresponds to a ring and each processor element (PE) except for the lowest layer PEs is connected to two PEs located at the immediate upper layer. The relation between layers is like binary-tree and the number of PEs at the n-th (from the top) layer is 2^n (see Fig. 1).

In [9], the PDGA is proposed on the assumption that it carries on the particular parallel machine model. The objective of this paper is to extend the PDGA to the hierarchical one and to investigate its effectiveness on ordinary loosely-coupled multiprocessor systems. Moreover, we show implementation of the hierarchical PDGA on an Ethernet-based parallel machine platform using MPI (message passing interface) [10].

In Sect.2, we describe the detail of the PDGA and propose the hierarchical PDGA. In Sect.3 we investigate the effectiveness of the hierarchical PDGA by comparing Cone with the PDGA and lattice topology with the hierarchical one. Moreover we show the implementation of the hierarchical PDGA on bus connection in Sect.4. Finally, We conclude this paper in Sect.5.

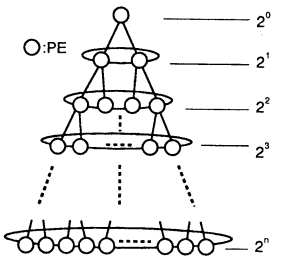


Figure 1: Cone topology

2 PDGA and its Application to Hierarchical Mechanism

In this section, we describe the detail of a parallel and distributed genetic algorithm (PDGA) on loosely-coupled multiprocessor systems. Since the isolated execution (no communication) of GAs is not effective [5], in our method cooperation among PEs, say transfer of chromosomes, occurs on the neighboring PEs (we say two processors are neighbor when a directed communication link connects them). We call chromosome-transfer *migration* and migrated chromosome *migrant*.

At the beginning of the execution, each PE generates its own chromosome set according to the GA parameters. Then each PE carries out genetic operations, crossover, mutation, and selection, on its population at each generation like ordinary GAs. Before the next generation, the migration takes place if its condition holds. We call this condition *migration condition*. The procedure for each PE is shown as follows:

```
0: procedure PDGA;
1: begin
2:    Prepare for genetic operation by a
      problem instance and parameters;
3:    Initialize a population;
```

```
4:    repeat
5:      Evaluate all the chromosomes;
6:      if the migration condition holds then
7:        Do the chromosome migration;
8:      end if;
9:      Select the chromosomes for the next
        generation by the roulette wheel way
        with the elite strategy;
10:     Create new chromosomes by applying
        crossover and mutation according to
        the crossover and the mutation rate;
11:   until the termination condition holds;
12: end;
```

In our previous study [9], we employed three standard topologies and our proposed Cone topology for the interconnection network of a parallel machine. The solution obtained from Cone topology was better than that from the others. A feature of Cone topology is hierarchical connection of ring. Consequently, its hierarchical topology brings out the effectiveness of migrant among PEs. We propose the hierarchical PDGA not to take account of the particular topology.

The hierarchical PDGA has to be organized by two PEs at least. Each layer has several PEs as a group and they are connected to specific neighboring PEs since many connections of other PEs occur the immature convergence in a chromosome set. Some PEs in the layer have a connection network to an immediate upper layer PE except for the lowest layer one. That is, a migrant emigrates from another PE located in the same or upper layer. The migration from upper to lower layer does not admit on the hierarchical PDGA.

3 Hierarchical PDGA on Ordinary Topology

Figure 2 and 3 show a torus and hypercube topology, respectively (four edges are omitted for being easy to see in Fig. 3). They are ordinary topology for the multiprocessor systems and each PE has several direct connection links. Many direct links affect the evolution by PDGA as immature convergence. We propose not to use all connection links of PE for avoiding the immature convergence, moreover to control active (migration) connections. We generate the hierarchical topology virtually in the previous two ordinary ones. Thick line means the active link and those lines compose the hierarchical topology in Fig. 2 and 3. That is, in Fig. 2 there are three layers in which the outside and small square show the lowest and middle layer, respectively. The center PE represents by itself the top layer in Fig. 2. Each layer connects by active links. Figure 3 has two layers, the big and small cube show the lowest and top layer, respectively. Two active links connects with two layers in Fig. 3.

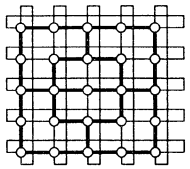


Figure 2: Torus topology

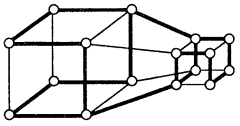


Figure 3: Hypercube topology

In this paper, we are not concerned with a dynamic interconnection network like a *omega*, *crossbar* network, and so on. Applying a dynamic interconnection network to the hierarchical PDGA is also an interesting topic.

We evaluate the solution quality generated by the proposed method to investigate the influence of modified standard topology to the hierarchical one. In this experiment, we solve by the PDGA eight benchmark instances [11, 12, 13, 14] of the multiple knapsack problem [15]. These problem instances are available from the OR Library [16] where problem instances with the optimum solutions are collected for several kinds of the combinatorial optimization problems. Our PDGA was run on a virtual multiprocessor system implemented on a Linux machine with Java. Throughout the experiment, the following parameters are used in each PE:

- generation span = 100
- population size = 3
- crossover rate = 1.0
- mutation rate = 0.0

In the experiment, we execute the hierarchical PDGA on the virtual multiprocessor systems to compare two topologies, Cone and lattice in which our proposed hierarchical PDGA runs, since the lattice topology is similar to torus and also makes hierarchical connections in itself. Figure 4 shows 6×6 lattice network and it is regarded as hierarchical topology of three layers: The outside processors in the lattice network compose a group of the lowest layer in the hierarchical topology, and the group of processors for the layer composes as the lattice goes inside, (see Fig. 4). We employ all connection links for the migration in this experiment.

Table 1 shows experimental results comparing Cone and lattice topology. Three columns, namely 16, 32, 64 in Cone, represent by percentage how often among fifty executions of the previous PDGA obtains the optimum solution for eight problem instances in Cone topology and

Table 1: Frequency to obtain the optimum

Problem	Cone		[%]	elite		[%]	random		[%]	random&elite		[%]
Instance	16	32	64	5×5	8×8	10×10	5×5	8×8	10×10	5×5	8×8	10×10
hp1	14	18	48	12	54	58	20	66	66	20	42	76
hp2	12	42	80	38	68	96	50	96	98	50	84	98
pb5	8	12	16	18	20	30	14	40	64	20	50	52
pb7	4	12	24	6	24	42	20	46	72	40	38	70
sento2	4	8	24	0	12	22	4	20	12	6	22	22
weing7	0	0	0	2	6	8	6	4	4	4	4	8
weing8	12	42	64	32	76	72	56	84	98	36	88	94
weish25	18	38	64	14	38	50	22	62	86	16	68	74

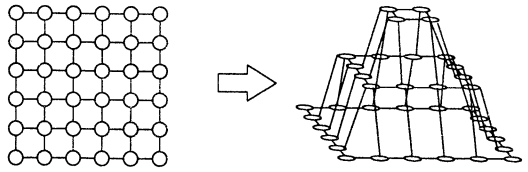


Figure 4: 6×6 lattice topology

three cases of the number of processors, say 16, 32, and 64. The value of remaining nine columns has the same substance and they are obtained from lattice topology by the hierarchical PDGA. Names of columns, *elite*, *random*, and *random&elite*, in Table 1 mean the type of migrants. The column of *elite* indicates that migrant is an elite chromosome which has the best fitness value in the population, *random* a randomly-selected chromosome, *random&elite* a randomly-selected chromosome in the same layer and an elite one for upper layer. These columns are divided into three inner columns according to the size of lattice topology (the number of processors).

From this table, we find that the results obtained from lattice topology is almost same or better than Cone. That is, we obtained better solution quality than torus from lattice topology by the hierarchical PDGA, since the solution quality obtained from Cone is better than that of torus from the previous experiment [9].

4 Hierarchical PDGA on Bus Connection

We propose and evaluate the implementation of the hierarchical PDGA on multiple PCs connected by bus-type network (e.g., Ethernet) in this section.

The hierarchical PDGA on bus is implemented by C++ with MPI [10]. Processes generated by MPICH have to be divided into several groups, that is, a process group corresponds to one layer and the leader of group communicates to the upper. Figure 5 shows a process group and its message passing on twenty-five processes. Each process group has the same number of processes and each process sends a migrant to neighbor in the group, that is, Each process communicates with another process in the same layer as the PDGA with a ring topology. A leader of each process group sends the migrant to its upper layer.

We investigate the solution quality and processing time of the hierarchical PDGA on Ethernet. The problem instances and GA parameters are the same as in the previous experiment. An elite chromosome is selected as a migrant transfered within the same group, and also one chromosome is chosen randomly in leader's population as a migrant to upper layer. We use 4 PCs (SMP Linux RedHat) connected with 100BASE-TX switching HUB. Each process carries out its own operation for 0.5 seconds.

Table 2 shows the experimental results comparing two problem instances, hp1 and weish25. The column, namely "group×layer" represents the size of process group and the number of layers. The values in column "GO time" and "Comm time" mean the average processing time for genetic operation and for communication at all process, respectively. The last column shows the average of fitness value of the final solution. From these results, we find out that communication time increases as the number of processes goes on increasing, that is, many processing delays which are the collision of packets, the blocking from another process are occurred on Ethernet network. Moreover, since the optimum values are 3418 (hp1) and 9939 (weish25), the lower improvement of obtained fitness values (see "fitness ave" columns in Table 2) is cased by much communication among processors. We find out that many delays of message passing and less migration are an important factor for the solution quality using the Ethernet.

5 Concluding Remarks

This paper considered the influence of the hierarchical PDGA. In the experiment, we compared Cone with lattice topology according to the fitness value. We found that our proposed hierarchical PDGA obtained almost

Table 2: Processing time and fitness average of hierarchical PDGA on bus connection

group×layer	hp1 (chromosome length=28)			weish25 (chromosome length=80)		
	GO time [%]	Comm time [%]	fitness ave	GO time [%]	Comm time [%]	fitness ave
4×4	20.67	79.33	3378.28	44.65	55.35	9767.60
5×5	16.89	83.11	3393.14	42.64	57.36	9824.60
6×6	13.46	86.54	3395.26	36.85	63.15	9827.78
7×7	11.54	88.46	3396.94	35.99	64.01	9832.56
8×8	10.64	89.36	3399.20	31.94	68.06	9833.90

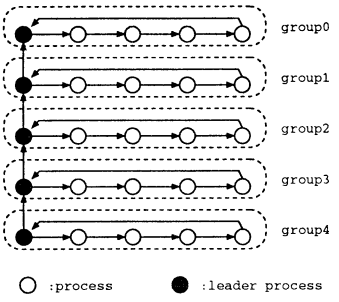


Figure 5: Hierarchical PDGA by MPI

same or better solution than Cone. Moreover, we investigated the processing time by the hierarchical PDGA on bus connection. From the experiment using Ethernet, the solution quality of the hierarchical PDGA did not improve very much. This was caused by the communication delays among processes. As future works, we analyze more detail issues about the delays among processors on bus connection, and develop the hierarchical PDGA on Ethernet using MPI. We also consider to apply the hierarchical PDGA to a dynamical interconnection networks.

References

[1] S.S.Patnaik,"Genetic algorithms: A survey," *IEEE Computer Society, Computer*, Vol.27 No.6, pp.17–26, 1994.

[2] L.Davis, *Handbook of genetic algorithms*, Van Nostrand Reinhold, 1991.

[3] C.C.Pettey, M.R.Leuze, "A theoretical investigation of a parallel genetic algorithm," *Proc. of the 4th International Conference on Genetic Algorithms*, pp.398–405, 1989.

[4] P.Spiessens, B.Manderick, "A massively parallel genetic algorithm implementation and first analysis," *Proc. of the 4th International Conference on Genetic Algorithms*, pp.279–286, 1991.

[5] J.Okech, M.Nakamura, K.Onaga, and S.Kyan, "A distributed genetic algorithm for the multiple knapsack problem using PVM," *Proc. of ITC-CSCC'97*, pp.899-902, 1996.

[6] H.Mühlenbein, M.Schomisch, J.Born, "The parallel genetic algorithm as function optimizer," *Proc. of the 4th International Conference on Genetic Algorithms*, pp.271–278, 1991.

[7] R.Sterritt, K.Adamson, M.Shapcott, et al., "A parallel genetic algorithm for cause and effect networks," *Proc. of the IASTED International Conference, Artificial Intelligence and Soft Computing*, pp.105–108, 1997.

[8] R.Tanese, "Distributed genetic algorithms," *Proc. of the 3rd International Conference on Genetic Algorithms*, pp.434–439, 1989.

[9] T.Matsumura, M.Nakamura, J.Okech, K.Onaga, "A parallel and distributed genetic algorithm on loosely-coupled multiprocessor systems," *IEICE Trans. Fundamentals*, vol.E81-A, No.4, pp.540–546, 1998.

[10] MPI forum, *MPI: A message-passing interface standard*, http://www.mpi-forum.org, 1995.

[11] S.Senyu and Y.Toyada, "An approach to linear programming with 0-1 variables," *Management Science*, vol.15, pp.196–207, 1967.

[12] H.M.Weingartner and D.N.Ness, "Methods for the solution of the multi-dimensional 0/1 knapsack problem," *Operations Research*, vol.15, pp.83–103, 1967.

[13] W.Shi, "A branch and bound method for the multi-constraint zero one knapsack problem," *J. Opl. Res. Soc.*, vol.30, pp.369–378, 1979.

[14] A.Freville and G.Plateau, "Hard 0-1 multiknapsack test problems for size reduction methods," *Investigation Operativa*, vol.1, pp.251–270, 1990.

[15] K.S.Back, T.Heitkotter, "The Zero/One Multiple Knapsack Problem and Genetic Algorithms," *Proceedings of the 1994 ACM Symposium of Applied Computation*, ACM Press, 1994.

[16] OR Library, URL http://mscmga.ms.ic.ac.uk/

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A study of Q-learning: Dynamic Structuring of Action Space Based on Genetic Algorithm

Kazuyuki Ito Fumitoshi Matsuno
Department of Computational Intelligence and Systems Science
Tokyo Institute of Technology
4259 Nagatsuta Midori Yokohama 226-8502 Japan
E-mail: kazuyuki@cs.dis.titech.ac.jp

Abstract

Reinforcement learning has recently been receiving much attention as a learning method for not only toy problems but also complicated systems for example robot systems. It does not need priori knowledge and has higher capability of reactive and adaptive behaviors. However increasing of action-state space makes it difficult to accomplish learning process. In the most of all previous works, the application of the learning is restricted to simple tasks with small action-state space. Considering this point, we present a new reinforcement learning algorithm: "Genetic Algorithm based action space structuring Q-learning". The algorithm is applicable to the systems that have large action space, for example a robot with many redundant degrees of freedom. To demonstrate the effectiveness of the proposed algorithm simulation of obstacle avoidance of a 30 links manipulator has been carried out. As the result, effective behavior has been obtained by using our proposed algorithm.

1 Introduction

Reinforcement learning has recently been receiving much attention as a learning method for not only toy problems but also complicated systems for example robot systems. It does not need priori knowledge and has higher capability of reactive and adaptive behaviors. However there are some significant problems in applying it to real robot tasks. Some of them are deep cost of learning and large size of action-state space.

In the Q-learning[1], increasing of size of action-state space makes it difficult to accomplish learning process. The application of the usual Q-learning is restricted to simple tasks with the small action-state space. It is difficult to apply the usual Q-learning to the real robot that has many redundant degrees of freedom and large action-state space. In order to apply the learning algorithms to the systems with the large action-state space, the various structural and dividing algorithms of the action-state space were proposed[2][3]. In the dividing algorithm, the state space is divided dynamically, however the action space is fixed and it is impossible to apply the algorithm to the task with large action space. In the classifier system that is a typical example for structural algorithms, "don't care" makes it possible to create general rules, however it causes the partially observable problem. And still more there is a problem of impossibility of coexistence of general rules with special rules. It was only applied to the large state space systems and not to large action space.

Considering these points, we present a new reinforcement-learning algorithm: "Genetic Algorithm based action space structuring Q-learning" which can solve the problems of the classifier system and be applied to the problem with the large state and action space. We assume that the world constitute a Markov division process. Under the assumption the algorithm is applicable to the systems that have large action space. This algorithm has two dynamics. One is the learning dynamics based on Q-learning, and the other is structural dynamics based on Genetic Algorithm. Genetic Algorithm is used to select effective actions from large action space. Q-table is structured from actions that are generated by Genetic Algorithm. Q-learning has been carried out and fitness of the genes is calculated from the reinforced Q-table. And again, actions are restructured using Genetic Algorithm. This cycle is repeated and effective behavior is acquired. In our proposed algorithm, state space does not structured directly, however under a condition that we propose, some part of space can be structured dynamically with structuring action space and the problem with large state and action space can be solved. Details of the condition are expressed in the section 3. The remarkable point of our proposed algorithm is the method of calculating fitness from the Q-table. To generate general action and special action simultaneously, we contrive the method. Another advantage of the proposed algorithm is the coexistence of general actions with special actions and exclusion of the partially observable problem caused by using

"don't care".

To demonstrate the effectiveness of proposed algorithm simulation of obstacle avoidance of a 30 links manipulator is carried out.

2 Learning Dynamics

2. 1 Outline

Proposed algorithm has two dynamics. One is a learning dynamics based on Q-learning and the other is a structural dynamics based on Genetic Algorithm. Fig. 1 shows the flowchart of our proposed algorithm.

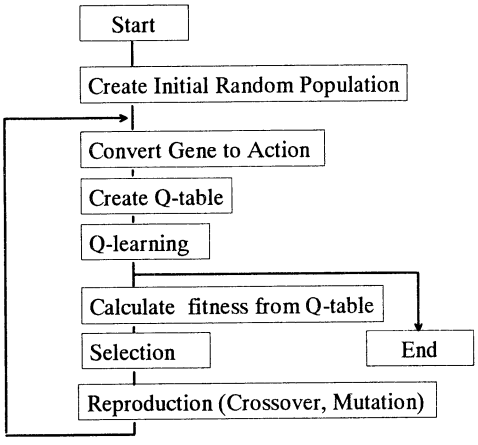


Fig. 1 Flowchart of proposed algorithm

Each action is expressed as a phenotype of genes and restructured by Genetic Algorithm. At first, an initial set of population is structured randomly, and the Q-table that consists of phenotype of initial population is constructed. The Q-table is reinforced using learning dynamics and the finesses of genes are calculated based on the reinforced Q-table. Selection and reproduction are applied and new population is structured. Repeating this cycle, effective behaviors are acquired.

2. 2 Cording

In this algorithm, each individual express the selectable action on the learning dynamics. It means that subsets of actions are selected and learning dynamics is applied to the subset. The subset of action is evaluated and new subset is restructured using Genetic Algorithm. The number of individual means the size of the subset and fewer size of subset make it impossible to accomplish the tasks and Genetic Algorithm does not work effectively. So the size of the population should be chosen as the large enough dependently on the complexity of the task. In our proposed algorithm, the number of actions that consist of the Q-table is reduced as learning is advanced. So our proposed algorithm works effectively even when the size of population is large.

2. 3 Consisting of Q-table

To reduce the redundancies of actions, the genes that have same phenotype are regarded as one action and the Q-table is consists of all different actions. And the size of the Q-table is decreased with the convergence of the obtained actions from structural dynamics using Genetic Algorithm.

In this algorithm, state space is not structuring dynamically. It is structured in advance. But by designing the action space suitably, some parts of state space can be structured dynamically. The way of the design is discussed in the section 3.

2. 4 Learning dynamics

The conventional Q-learning[1] is employed as a learning dynamics. The dynamics of Q-learning is written as follows

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(r(s,a) + \gamma \max_{a'} Q(s',a')) \quad (1)$$

where
s : state, a : action, r : reward
α : learning rate, γ : discount rate

2. 5 Calculating of fitness

The fitness of genes is calculated at two steps. The first step is regulation of the Q-table and the second step is calculating of the fitness from the regulated Q-table.

At first, we calculate the maximum and minimum value of the state as follows.

$$V_{\max}(s) = \max_{a' \in a}(Q(s,a'))$$
$$if\ V_{\max}(s) < 0\ then\ V_{\max}(s) = 0$$
$$V_{\min}(s) = \min_{a' \in a}(Q(s,a'))$$
$$if\ V_{\min}(s) > 0\ then\ V_{\min}(s) = 0$$

Then Q' of the regulated Q-table is given as follows

$$if\ Q(s,a) \geq 0\ then$$
$$Q'(s,a) = \frac{1-p}{V_{\max}(s)} Q(s,a) + p$$
$$if\ Q(s,a) < 0\ then$$
$$Q'(s,a) = -\frac{p}{V_{\min}(s)} Q(s,a) + p$$

where p is the constant value which means the ratio of reward to penalty.

Next, we fix the action to a_i and sort $Q'(s,a_i)$ according to their value from high to low for all state, and we define it as the $Q'_s(s,a_i)$ and repeating the operation for all actions. For

example $Q'_s(1,a_i)$ means the maximum value of $Q'(s,a_i)$ and $Q'_s(N_s,a_i)$ means the minimum value of $Q'(s,a_i)$, where N_s is the size of state space.

In the second step, we calculate the fitness. The fitness of gene whose phenotype is a_i is given as follows

$$fit(a_i) = w_1 \frac{Q'_s(1,a_i)}{1} + w_2 \frac{Q'_s(1,a_i)+Q'_s(2,a_i)}{2} + \cdots + w_{N_s} \frac{Q'_s(1,a_i)+Q'_s(2,a_i)+\cdots+Q'_s(N_s,a_i)}{N_s} \quad (2)$$

$$= \sum_{j=1}^{N_s} \left(w_j \frac{\sum_{k=1}^{j} Q'_s(k,a_i)}{j} \right)$$

where w_i is a weight which decides the ratio of special actions to general actions.

The fitness defined in (2) has the three important points. The first point is the regularization of the state value of the Q-table. In the Q-learning, the value of the state that is closer to goal state is higher. So if the fitness is calculated from unregulated Q-table, the selected actions at the state that close to the goal are evaluated as a high values. And the actions that are selected near the start state are evaluated as a low values and they are extinguished. But to accomplish the task, a series of actions are needed. So the regularization of state value of the Q-table is important.

Second point is the handling of penalty. At the Q-learning, the penalty that has negative value is employed. But the fitness of Genetic Algorithm should be positive, so the conversion of penalty to the fitness is necessary. At the proposed method, the positive value of the Q-table is converted to the value from p to 1.0 and the negative value converted to the value from 0 to p. We can choose the rate of the reward to penalty by selecting the value of p.

Third point is the method of calculation of the fitness. The first term of the equation (2) means the maximum value of the action. When w_1 is chosen as a large value, the action that is effective in the special state is evaluated as a high credit, and the special actions are generated by Genetic Algorithm. The last term of (2) implies the mean value of the action. And when w_{Ns} is chosen as a large value, the action that is effective in the various states is evaluated as a high credit and general actions are generated. Selecting the weight, we can choose the ratio of the special actions to general actions. And in the proposed algorithm, the special actions and the general actions are evaluated as a high credit simultaneously, so they can coexist. And our proposed algorithm does not have "don't care" symbol, so the problems that caused by using "don't care" symbol do not exist.

2. 6 Selection and Reproduction

Various methods of selection and reproduction that have been studied can be applied to our proposed algorithm. The method of the selection and reproduction should be chosen for each given task. In this paper the method of the selection and reproduction is not main subject so the conventional method is used.

3 Dynamic structuring of state space

The dynamic structuring of the state space can be employed to our proposed algorithm under the following conditions.

The state space can be divided into two states as an interior state and an exterior state. The interior state can be controlled directly, for example a joint angle of the manipulator. The exterior state is like a state of environment, which cannot be controlled directly for example a position of the obstacle.

By designing the actions as the desired interior state, the dynamic structuring of the interior state space is possible. The action that is selected at one step past means the present interior state, so the interior state space that can be transited is equivalent to the action space that is selectable in the present learning dynamics.

In our proposed dynamics the actions that can be selected is structured dynamically, and by designing the state space of the Q-table as the selectable actions, the state space of the Q-table is structured dynamically as the action is structured.

4 Simulation

To demonstrate the effectiveness of the proposed algorithm, a simulation of obstacle avoidance of a 30 links manipulator is carried out.

4. 1 Task

We apply the proposed algorithm to the problem of the obstacle avoidance using a 30 joints manipulator. Let us define the origin and coordinate as shown in Fig. 3. The origin means the fixed end of the manipulator and the first joint angle is the absolute angle with respect to the x-axis. The other joint angles are relative angles to the next links and we define the straight state is a 0[deg] and counterclockwise is positive. The goal of the task is moving the top of the manipulator to the interior of desired circle with avoiding the obstacle. The length of each link of the manipulator is 0.05[m] and the initial attitude is the straight line on the x-axis. The obstacle is a

circle whose radius is 0.1[m]. Its center moves to the counterclockwise on the circular orbit whose center is (0.5, 0.5) and radius is 0.3[m] by constant speed. And the initial position on the circle is random. The goal position is the 0.1[m] radius circle and the center is (0.5,0.5) and does not move.

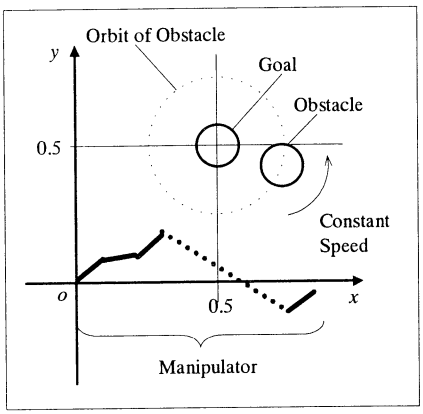


Fig. 3 Task of simulation

4. 2 Simulation model of the manipulator

We regard that the actuator is the stepping motor and the angle and angular velocity can be controlled. We introduce following assumptions. All joint is moved to the desired angle by the same constant angular velocity. And when the joint reaches the desired angle, the joint is stopped. And when all joint angles reach the desired angle, the manipulator is stopped.

4. 3 Formation of Genetic Algorithm

The dynamics of Genetic Algorithm of the proposed algorithm is composed as follows.

At first we describe the cording. We define the action as the desired angles of the joints. And the actions are corded as the gene. One action expresses the all joint angles related to the one shape of the manipulator. One action is corded one chromosome and the chromosome has a same number of genes as the number of joints. One gene expresses the angle of one joint. One gene has 9 characters that express the angles from –40[deg] to 40[deg] every 10 degrees. The number of individuals is 200. The roulette selection is employed. The probability of the crossover is 0.2 and one-point crossover is employed. The probability of mutation is 0.001. And 10 times reproduction is carried out. The chromosome whose phenotype shows the cross shape of the manipulator is regarded as lethal genes and is extinguished.

4. 4 Formation of Q-learning

The action space consists of the phenotypes of the generated genes. The state space consists of interior states and exterior states. The interior states are composed of the initial state and the states that can be transited by generated actions. Thus the interior state is generated dynamically with actions which is generated. The external states consist of the positions of obstacle. The positions of obstacle are divided into 20 parts.

When the top of manipulator reaches the goal, the value of 100 is given as a reward. When any part of the manipulator touches the obstacle, the value of –50 is given as a penalty. The roulette selection using Boltzmann distribution is employed. The learning rate is 0.5 and discounting rate is 0.9. The number of trials of each learning dynamics is 1000 times.

4. 5 Acquired behavior

Fig. 3 shows the acquired behavior. We can find that the effectiveness behavior was acquired and the task is accomplished.

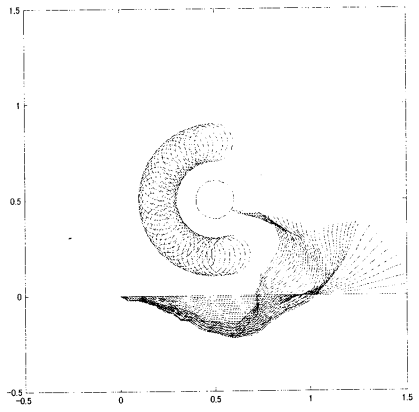


Fig. 3 Acquired behavior

5 Conclusion

We propose the new reinforcement algorithm. To demonstrate the effectiveness of proposed algorithm simulation of obstacle avoidance of a 30 links manipulator has been carried out. As the result, effective behavior has been obtained by using our proposed algorithm. In the proposed algorithms, action space and interior state space were generated adaptively by Genetic Algorithm, so it is effective to the large action and interior state space systems.

Reference

[1] C. J. C. H. Watkins, P. Dayan, "Technical note Q-Learning," *Machine Learning*, Vol. 8, pp 279-292, 1992

[2] R. Munos, J. Patinel, "Reinforcement learning with dynamic covering of state-action space: Partitioning Q-learning," *Proc. of the 3rd Inter. Conf. on Simulation of Adaptive Behavior*, pp. 354-363, 1994

[3] M. Dorigo, H. Bersini, "A Comparition of Q-Learning and Classifier Systems," *Proc. of the 3rd Inter. Conf. on Simulation of Adaptive Behavior*, pp. 248-255, 1994.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Interaction mechanism between DNA and substrates clarified based on molecular dynamics

Yuto Komeiji
National Institute of Advanced Industrial Science & Technology (AIST)
Tsukuba, Japan. 305-8562
komeiji@nair.go.jp
http://www.etl.go.jp/~komeiji/

Abstract

In this paper, recent molecular dynamics studies on polynucleotides (DNA/RNA) and polynucleotide/peptide molecular complexes are reviewed.

1 Introduction

DNA or RNA binding by polypeptides plays important roles in a variety of biomolecular systems, for instance, gene expression, replication, recombination, and so on.

Physico-chemical basis of DNA/RNA recognition has not been well understood, but the availability of the vast of 3D structures of DNA/RNA-polypeptide complexes have enabled us to characterize their conformational dynamics by means of molecular dynamics (MD) and related simulation methods. I would like to review the recent advancement in MD of the polynuculetides (DNA/RNA) and in that of the complexes.

2 MD of Polynucleotides

MD is a method of molecular simulation to compute dynamic and thermodynamic properties of a system by solving Newton's equation of motion of all the constituent atoms of the molecule [1]. MD was originally developed and used to solve statistical problems of hard sphere systems in 1950s, and the first MD simulation of a protein appeared in 1977 [2]. Several reports of MD of polynucleotides including DNA appeared in the literature in 1980s; however, realistic simulations of nucleotides have become available to us only in the mid-1990s.

A polynucleotide is a highly charged chain molecule sensitive to the environment and treatment of the electrostatic interaction, and hence is usually more difficult to simulate than a polypeptide is. During 1980s it was common to simulate a biomolecule without explicit solvent by using an electrostatic cutoff. Also, the force field parameters based on low precision MO calculations such as STO-3G were used. These MD methods were not bad, if not ideal, for simulation of small, globular proteins. However, they were not suitable for the polynucleotides.

The situation was changed by the mid-1990s. The dramatic improvement in computer hardware has made it possible to incorporate the explicit solvent molecules to simulation. A new force field parameter set was developed based on high precision MO calculations (6-31G*, [3]). Several algorithms to compute the electrostatic interaction without the cutoff have been introduced to MD of biomolecules, among which the Particle Mesh Ewald method (PME, [4]) has been the most frequently used.

Thus, the inclusion of the explicit solvent, the new force field, and the 'no cutoff' methods have promoted realistic MD simulations of the polynucleotides (see reference [5] for a recent review). MD simulations of free DNA or RNA as long as several nanoseconds or several tens of nanoseconds are frequently performed. It is now possible to reproduce B<->A conformational transitions depending on the solvent environment [6-9]. A spine of water along the minor groove has been detected in many of MD simulations [10-12].

In summary, the improvements in MD protocol and computer hardware have led to realistic modeling of the polynucleotide

molecules and their environment.

3 MD of DNA/RNA-Polypeptide complexes

3.1 General View

In the previous section, I reviewed briefly the recent advancement in the simulation of the polynucleotides, and mentioned that the explicit solvent, the 'no cutoff' of the electrostatic interaction, and the new force field were necessary. Now that the realistic simulations of DNA and RNA are available, then it is natural to think of DNA (RNA)-polypeptide complexes as the next target of the study by the new MD protocol.

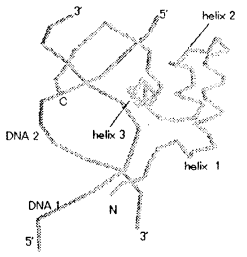
To date, the number of MD simulations of such complexes performed by the new protocol is limited. Nevertheless, a few examples have shown general applicability of MD to analysis of nucleotide recognition by polypeptides.

Schulten's group performed MD simulations of several kinds of DNA-polypeptide complexes [13-15]. In their study of Glucocoriticoid receptor-DNA complexes [13], the protein induced DNA bending was observed. Also, a more favorable interaction was seen between the protein and the consensus DNA sequence than the nonconsensus sequence. Similar results were obtained in the case of the Hormone-receptor [14] and Estrogen receptor [15]. In the latter case, water-mediated hydrogen-bonds were found to be important in recognition of the consensus sequence. The importance of water-mediated hydrogen-bonds were also demonstrated in the simulations of the homeodomain-DNA complex by Billeter et al. [16] and in the simulation of the Tryptophan-repressor/operator complex by Suenaga et al. [17]. RNA recognition is also studied. Hermann and Westhof [18] studied the effect of ionic strength on RNA-protein interface, and showed that salt-induced destabilization of the complex was clearly reproduced.

3.2 MD of Hin/DNA complex

MD is also useful in analyses of the conformational dynamics of DNA/peptide complexes. For the rest of this article I would like to mention our MD study of the Hin-recombinase/DNA complex, as an example of characterization of the conformational change upon complex formation [19, 20].

Fig. 1 Structure of Hin/DNA complex



It is a molecular complex formed by the C-terminal 52mer peptide of the Hin-recombinase protein and the half site hixL DNA (13 bp). The crystal structure of the complex ([21], Fig. 1) shows that the peptide recognizes the DNA by its helix-turn-helix motif inserted in the major groove and by two termini inserted in the minor groove. The hixL DNA adopted a conformation close to the canonical B form; however, the minor groove was wider than expected from its AT-rich sequence. Biochemical experiments showed that the deletion of the two N-terminal residues resulted in complete loss of the binding activity [22], while deletion of the eight C-terminal residues resulted only in a reduced affinity [23].

We performed MD simulations based on the crystal structure, and tried to examine if the conformational dynamics revealed by MD could explain the structural and biochemical properties of this molecular complex.

Our inhouse software PEACH ver. 2 [24] was used throughout the study. Three simulations were performed: MD-pep (simulation of the free peptide), MD-dna (the free DNA), and MD-com (the complex). In each simulation the solute molecule(s) were placed in an explicit solvent box with counter ions to neutralize the system. See

Fig. 2 for the initial configuration of MD-com. Each simulation was continued for 2.5 ns at 300 K, and 0.5-2.5 ns trajectory was used for analyses. See [19, 20] for further details.

Fig. 2 Initial configuration for MD-com

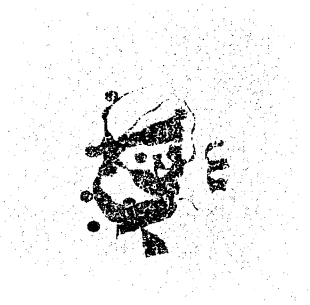
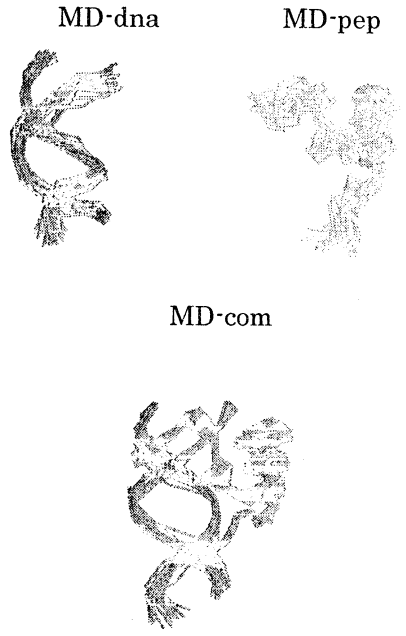


Fig. 3 Superimposed snapshots of simulated trajectories



The results of MD are best illustrated by the superimposed configurations of the simulated trajectories (Fig. 3).

First, compare the configuration of the peptide in MD-pep and MD-com. The fluctuations of the both termini were large in MD-pep but small in MD-com. It is obvious that the N-terminal region is firmly attached to DNA, though the contact surface was small. The C-terminal residues were loosely associated with the minor groove. These MD results were consistent with the essential role of N-terminus and auxiliary role of C-terminus in DNA recognition.

The difference in DNA conformation between MD-dna and MD-com was small but significant. The minor groove was narrower in MD-dna, suggesting that the intruding peptide widens the groove upon complexion. Though I do not mention here, a detailed analysis showed that the conformation in MD-com was somewhat close to A form, but that in MD-dna was completely the canonical B form [20].

In summary, the MD simulations showed the induced fit of both DNA and the peptide upon complexion. Also, the conformational dynamics revealed by the simulations were consistent with the available biochemical data.

4 Perspective

I have reviewed the recent advances in MD simulations of the polynucleotides and polynucleotide/peptide complexes. This field is still advancing rapidly; for example, the free energy calculation upon complex formation of a RNA binding protein was reported by Reyes and Kollman [25]. Thus, quantitative estimation of the binding affinity has become possible. We expect that MD simulation and related computational methods will play an important role in understanding and prediction of DNA recognition.

References

[1] Komeiji Y, Uebayasi M, Nagashima U (2000), REVIEW: Molecular Dynamics Simulation of Biological Molecules (1) Methods (in Japanese), J. Chem. Software 6, 1-36.

[2] McCammon J, Gelin B, Karplus M (1977), Dynamics of folded proteins, Nature 267: 585-590.

[3] Cornell W, Ciplak P, Layly C, et al. (1994), A second generation force field

for the simulation of proteins, nucleic acids, and organic molecules, J. Am. Chem. Soc. 117, 5179-5197.
[4] Darden T, York D, Pedersen L (1993), Particle mesh Ewald: An NlogN method for Ewald sums in large systems, J. Chem. Phys. 98, 10089-10092.
[5] Beveridge D, McConnell K (2000), Nucleic acids: theory and computer simulation, Y2K, Curr. Opin. Struct. Biol. 10: 182-196.
[6] Cieplak P, Cheatham T, Kollman P (1997), J. Am. Chem. Soc. 119, 6722-6730.
[7] Cheatham T, Kollman P (1997), Structure 5, 1297-1311.
[8] Sprous D, Young M, Beveridge D (1998), Molecular dynamics studies of the conformational preferences of a DNA double helix inwater and an ethanol/water mixture: theoretical considerations of the A<->B transition, J. Phys. Chem. B 102: 4658-4667.
[9] Jayaram B, Sprous D, Young M, et al. (1998), Free Energy Analysis of the conformational preferences of A and B forms of DNA in solution, J. Am. Chem. Soc. 120: 10629-10633.
[10] De Winter H, Lescrinier E, Van Aerschot V et al. (1998), Molecular dynamics simulation to investigate differences in minor groove hydration of HNA/RNA hybrids as compared to HNA/DNA complexes, J. Am. Chem. Soc. 120, 5381-5394.
[11] Feig M, Pettitt B (1999), Modeling high-resolution hydration patterns in correlation with DNA sequence and conformation, J. Mol. Biol. 286, 1075-1095.
[12] Castrignano T, Chellemi G, Desideri A (2000), Structure and hydration of BamHI DNA recognition site: a molecular dynamics investigation, Biophys. J. 79, 1263-1272.
[13] Bishop T, Schulten K (1996), Molecular dynamics study of Glucocorticoid receptor-DNA binding, Proteins 24, 115-133.
[14] Bishop T, Kosztin D, Schulten K (1997), How Hormone receptor-DNA binding affects nucleosomal DNA: the role of symmetry, Biophys. J. 72: 2056-2067.
[15] Kosztin D, Bishop T, Schulten K (1997), Binding of the Estrogen receptor to DNA: the role of waters, Biophys. J. 73, 557-570.
[16] Billeter M, Guntert P, Luginbuhul P (1996), Hydration and DNA recognition by Homeodomains, Cell 85, 1057-1065.
[17] Suenaga A, Yatsu C, Komeiji Y, et al. (2000), Molecular dynamics simulation of trp-repressor/operator complex: analysis of hydrogen bond patterns of protein-DNA interaction, J. Mol. Struct. 526: 209-218.
[18] Hermann T, Westhof E (1999), Simulations of the dynamics at an RNA-protein interface, Nat. Struct. Biol. 6, 540-544.
[19] Komeiji Y, Uebayasi M (1999), Molecular Dynamics Simulation of the Hin-Recombinase-DNA Complex, Mol. Simul. 21: 303-324.
[20] Komeiji Y, Uebayasi M (1999), Change in conformation by DNA-peptide association: Molecular dynamics of the Hin-recombinase-hixL complex, Biophys. J. 77: 123-138.
[21] Feng J, Johnson R, Dickerson R (1994), Hin recombinase bound to DNA: the origin of specificity in major and minor groove interactions, Science 263: 348-355.
[22] Sluka J, Horvath A, Glasgow A, et al. (1990), Importance of minor groove contacts for recognition of DNA by the binding domain of Hin recombinase, Biochemistry 29: 6551-6561.
[23] Mack D, Sluka J, Shin J, et al. (1990), Orientation of the putative recognition helix in the DNA binding domain of Hin recombinase complexed with the hix site, Biochemistry 29, 6561-6567.
[24] Komeiji Y, Uebayasi M, Takata R, et al. (1997), Fast and accurate molecular dynamics simulation of a protein using a special purpose computer. J. Comp. Chem. 18: 1546-1563.
[25] Reyes C, Kollman P (2000), Structure and thermodynamics of RNA-protein binding: using molecular dynamics and free energy analyses to calculate the free energies of bind and conformational change, J. Mol. Biol. 297: 1145-1158.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Self Organization in Ecosystem

T. Shimada*, S. Yukawa, and N. Ito
Department of Applied Physics
The University of Tokyo
Bunkyo-ku, Hongo, Tokyo 113-8656, Japan

Abstract

Ecosystem, especially food web, is essentially characterized as a many-body system which is interacting each other under the limitation of the energy and resources[1]. We introduce the coevolutional population dynamics model for food webs which contains energy conserving interaction, energy dissipation, and rules for changing the degree of the freedom(extinction and mutation). In this model, diversity of the system increases spontaneously. The statistical properties of the system such as the distribution of the life time of the species will be also discussed in the talk.

1 introduction

Since the Lotka-Volterra equations were proposed, population dynamics has occupied an important position in the study of ecosystems. Although there are many variations of Lotka-Volterra type models, these can be generalized to the form as follows[2, 3],

$$\frac{dx_i}{dt} = c_i x_i + \sum_j a_{ij} x_i x_j. \qquad (1)$$

Where x_i, a_{ij}, c_i represent the population density of the ith species, interaction coefficient between ith and jth species, intrinsic growth rate, respectively. The analysis of this population dynamics and its linearized model by R.May and others in 1970s yielded a remarkable harvest. That is the fact that complex systems with strong interactions are merely stable, which is the opposite to the belief of that time[4, 5, 6]. However, it has also become known that the argument of May *et.al.* is too strict for the middle-sized systems with moderate interactions. It is also found that the exceptionally large or strongly interacting systems can be constructed by the trial-and-elimination scheme[7, 8, 9].

*shimada@acolyte.t.u-tokyo.ac.jp

Let us focus on the food web. Food web is the trophic links in a ecosystem. Therefore, one should treat not the population but the resource or energy in population dynamics. As mentioned below, this brings more difficulty to the one tries to make a model of food web. Although there are many type of models for food web[10, 11, 12], there has not been such model in which system is self-organized from the population dynamics and the trial-and-elimination.

2 model

Let us start to construct the model of food web from the generalized Lotka- Volterra equations (eqs.(1)). To give the system to have a chance to be stable, following the May *et.al*'s arguments, we set most of the interactions to be 0. The average number of the interaction per species is represented as m. Since interactions here are limited to the prey-predator relations, we also define the interaction coefficients as antisymmetric($a_{ij} = -a_{ji}$). Note that the *population* x_i actually refers the energy or resource in the biomass of the species. It's obvious that the sum of the *population* is conserved by the prey-predator interactions. We then decide that there is a unique *plants* which has a positive c_i and others are *animals* which have negative growth rate. These condition correspond to the production of the energy in plants and energy dissipation by metabolism in animals. The trial-elimination scheme is then realized as the adding the rules of random mutation and extinction. New species which has totally random m interactions comes randomly. Species whose population became very small are extinct and eliminated from the system.

Despite the expectation that the system grows to the rich structure spontaneously, it turns out that this naive modification doesn't work. Systems cannot yields the diverse structure and tend to collapse to the poor structure by the mutation and extinction. The main reason of the failure is the difficulty to de-

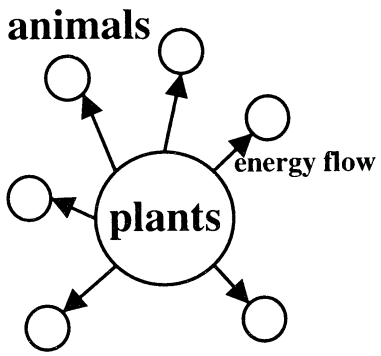


Figure 1: An example of trivial star-like structure which is yielded from simple Lotka-Volterra type interactions. System tends to collapse to this structure otherwise totally dies out.

cide suitable value of the interaction coefficients(a_{ij}). Since the interaction coefficients have a dimension of time per energy density, it must distribute over a quite wide range. For example, interaction between cows and grass would be represented much far larger than the one of eagles and rats. One solution is deciding the proper range for the interaction coefficients by hands according to the trophic levels. However, when we adopt this scheme, we must dispose the emergence of the trophic levels and the variety of the life styles.

Here we introduce another way; that is, rescaling of the interaction coefficient. In the Lotka-Volterra eqs., the rate of the preying per unit predator is linear function of a population of the prey. We modify this preying rate to the function of the ratio of the populations of the prey(j) and the predator(i) as

$$a_{ij}x_j \to a_{ij}\left(\frac{x_j}{x_i}\right)^{\lambda}, \quad (\lambda \in (0,1)) \tag{2}$$

Where the condition of the λ corresponds to the saturation of the preying rate. This preying rate(2) yields the interaction term of

$$a_{ij}x_i^{1-\lambda}x_j^{\lambda}. \tag{3}$$

Although the every λ also can be vary like a_{ij}, in this talk we set the parameter λ uniformly to 0.5 for simplify. The metabolism rate of the animals are also uniformly chosen to 1. After that, equations of the motions of our model are,

$$\begin{aligned} &\text{"plants"} \quad : \frac{dx_1}{dt} = Gx_1(1-x_1) + \textstyle\sum_{j\neq 1} a_{1j}\sqrt{x_1x_j} \\ &\text{"animals"} \quad : \frac{dx_i}{dt} = -x_i + \textstyle\sum_{j\neq i} \sqrt{x_ix_j} \end{aligned} \tag{4}$$

$$(a_{ij} = -a_{ji} \in (-1,1), \quad G > 0, \quad 0 < \lambda < 1),$$

where G denotes the growth rate of the plants. The condition $a_{ij} \in (-1,1)$ comes from the requirement that all the animals cannot survive when the total population of the prey is smaller than the population of themselves. This model has additional rules to change the degree of the freedom of the system as follows.

Additional Rules:

- *Extinction*: If energy of a species $i(x_i)$ becomes to 0 or less, that species extinct and the ith degree of the freedom is eliminated. In addition to this rule, instant extinction rule is applied to the species which is completely isolated from others.
- *Mutation(Invasion)* : A new species comes into the system randomly at time. Initial energy of the new comer is chosen to be very small(10^{-8}). Number of the interactions are decided randomly in the range of $(1, 2m)$, here m is the average number of interactions per a species. Amplitudes of the interactions(a_{ij}) are also randomly decided from certain distribution ρ. which is defined in $(0, 1)$.

It is worth stressing that we don't require the threshold for extinction. In Lotka-Volterra type models populations of the unmatched species decrease exponentially therefore threshold is necessary to cause the extinction[3, 9]. In contrast, $a_{ij}x_i^{1-\lambda}x_j^{\lambda}$ type interactions cause the algebraic decay of the unmatched species, so the orbits of the unmatched species decrease to 0 in finite time. The irreversibility of the extinction is represented naturally as the breaking of the uniqueness of the solution at $x_i = 0$.

3 results

Let us investigate the model mentioned above by numerical simulation. The parameters in this simulation are chosen to $G = 50, m = 5$, respectively. At the initial time there is plants alone in the system. This time system's diversity gradually grows by the mutation(invasion) although most of the new-commers fails to survive(Fig.3). To get the structural information, we classify the species by the trophic level. Trophic level is defined as the minimum distance to the plants. In other words, trophic level of the species who eat the plants is one, and trophic level of the species who don't eat plants and prey the species whose trophic

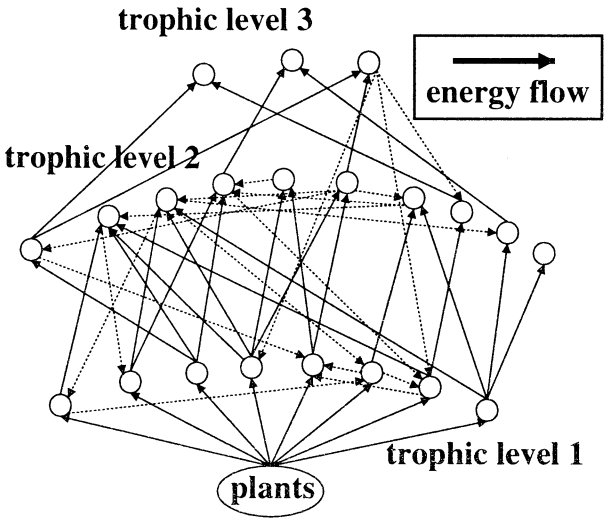


Figure 2: An example of the self-organized food web with rich structure. Species are represented as circles. Each arrow represents the preying interaction. There are 21 species and 3 tropic levels.

level is one is two and so on. In Fig.3 we can see the development of the food web in trophic level. Let us next see the long-time behavior of the system. System's diversity grows to infinite when the average number of interactions(m) and the strength of interactions(decided by ρ) are smaller than the value made by May *et. al.* (Fig.4). In the case that interactions are strong, the systems diversity goes on fluctuating and the power spectrum of the number of the species shows the $1/f$ distribution. In the latter case, the distribution of the life-time of the species shows the power-law tail(Fig.5).

4 Summary

We have introduced the simple model of the food web. In this simple model, system self-organizes to the rich structure by modifying the interaction as we have introduced. Systems with weak interactions grows to infinite. Systems with relatively strong interactions continue growing and shrinking.

References

[1] H. Hirata, J. theor. Biol. **162**, 187 (1993).

[2] P. J. Taylor, J. theor. Biol. **135**, 543 (1988).

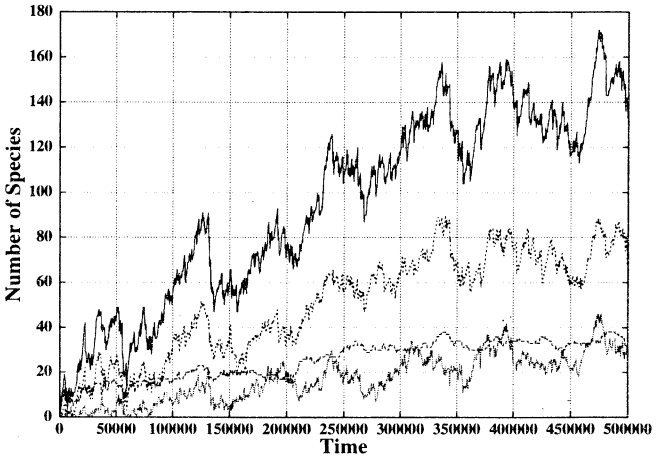


Figure 3: Time development of the number of the species in the system. Mutation rate is 0.1 per time. Four lines represents the number of; total species, species in the second trophic level, species in the first trophic level, species in the third trophic level, respectively in the order of top to bottom.

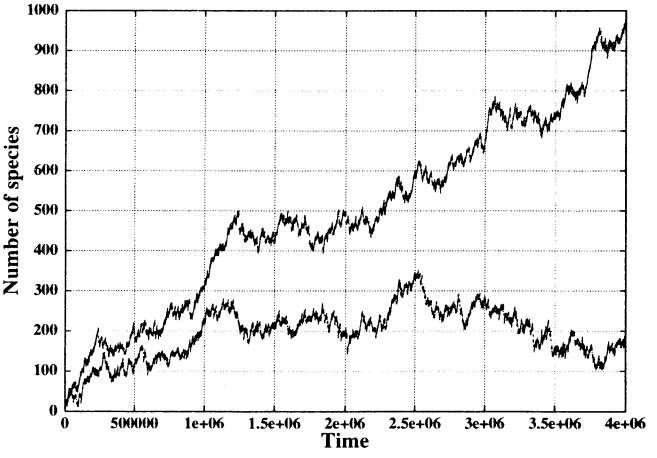


Figure 4: Long time behavior of the system. Upper line is the system with the weak interactions and lower line is the system with strong interactions.

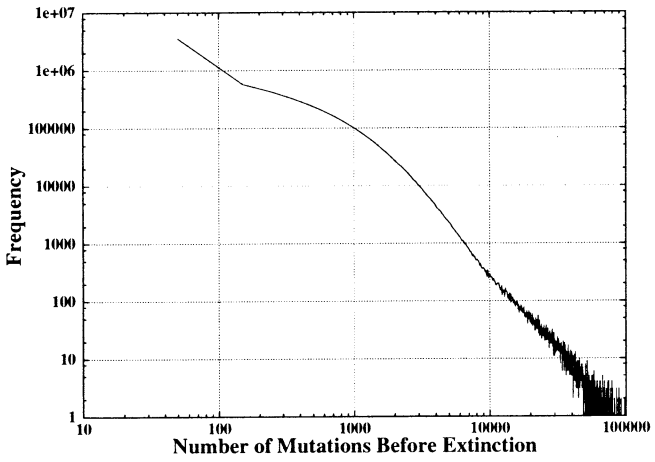


Figure 5: Distribution of the 'life-time' of species. Horizontal axis is the number of the mutations which a species has endured before it becomes extinct. In large number region, it shows the power law behavior.

[3] K. Tokita and A. Yasutomi, Phys. Rev. E **60** 842 (1999).

[4] R. M. May, Nature **238**, 413 (1972).

[5] A. Roberts, Nature **251**, 607 (1974).

[6] M. E. Gilpin, Nature **254**, 137 (1975).

[7] K. Tregonning and A. Roberts, Nature **281**, 563 (1979).

[8] A. Roberts and K. Tregonning, Nature **288**, 265 (1980).

[9] P. J. Taylor, J. theor. Biol. **135** 569 (1988).

[10] P. Bak and K. Sneppen, Phys. Rev. Lett. **71**, 4083 (1993).

[11] D. Stauffer and N.Jan, Int. J. Mod. Phys. C **11**, 147 (2000).

[12] R. J. Williams and N. D. Martines, Nature **404**, 180 (2000).

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Cyto-fluid Dynamic Theory of the Origin of Base, Information, and Function

Ken NAITOH

Yamagata Univeristy, 4-3-16 Jyonan, Yonezawa-shi, Yamagata, 992-8510 Japan
k-naito@dip.yz.yamagata-u.ac.jp

Abstract

The origin of transfer RNA (tRNA) molecules, that is, the origin of information and function, is clarified on the basis of the first principle of continuum mechanics and bio-informatics. It should be stressed that the ladder of asymmetry related to the artistic ratios of about 2:3, such as the golden and silver ratios, is the essential prerequisite for the auto-catalytic production of tRNA, the two-dimensional clover structure.

Introduction

Why do DNA and RNA mainly consist of five bases in nucleotides, that is, adenine (A), guanine (G), thymine (T), cytosine (C), and uracil (U) ? What kind of grammar can generate the complex clover structure of transfer RNA molecule (tRNA) ? These questions are related to the origin of biological information and function. An essential answer to these questions is presented here on the basis of continuum mechanics.

First, we show that a lot of asymmetric ratios of around 2:3, such as the golden and silver ratios, can be observed in actual living systems. The size ratio of purine bases and pyrimidine bases, for example, is about 2:3.

Next, an explanation is given to the question of why the 2:3 ratio is so pervasive in the densities of nucleic acids. Then, we propose a natural auto-catalytic reaction mechanism for generating one-dimensional strings containing the 2:3 ratio and also for generating the two-dimensional clover structure of tRNA.

An essential principle for creating biological order by integrating information, that is, the key prerequisite for the origin of biological information and functions is made clear.

Asymmetry Observed in Living Systems (Naitoh, 1999b, 2001)

We will begin by examining the asymmetric ratios, such as the golden and silver ratios around a ratio of 2:3, which are actually observed in several classes and parts of living systems.

(1) Nucleic Acids (Watson et al. 1987): DNA and RNA molecules are mainly constructed of five molecules, namely, adenine (A), guanine (G), thymine (T), cytosine (C), and uracil (U). The A-T pair has "two" hydrogen bonds and the G-C pair has "three". The nitrogen-containing bases in pyrimidines and purines have an asymmetric size ratio of about 2:3. This is because the main structure of purines A and G have two carbon-nitrogen rings, while pyrimidines T, C, and U have only single ring. The size ratio of double carbon-nitrogen rings in purine bases and single rings in pyrimidine bases is about 1.600, which are L_A / L_T, L_G / L_C, and L_A / L_U (Fig. 1). This value is close to the golden ratio (about 1.618). "Two" types of purines and "three" types of pyrimidines mainly exist in DNA and RNA. Chargaff data reveal that the A and G molecules in human beings, cattle, fowl, salmon, wheat, yeast, virus, and E.coli exist in ratios of 40% and 60%, respectively. Moreover, the diameter of DNA is about 23.7 angstroms, while one pitch of DNA is about 33.2 angstrom. This is also close to the 2:3 ratio.

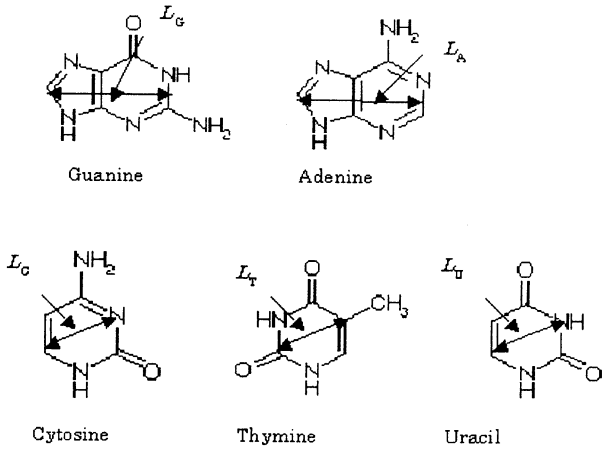


Fig. 1 Nitrogenous bases.

(2) Amino Acids: The number of all possible codons encoded by an array of three nucleic acids is sixty-four. Forty-four codons among them are degenerate. The numbers of degenerate codons and all possible codons have an asymmetric ratio of about 2:3.
(3) Ribosome : The numbers of proteins included in two sub-units of E. coli ribosomes are 21 and 34.
(4) Size Ratio of Child Cells : It is also known that, when a cell divides, the size ratio of two resulting child cells is nearly equal to 2 : 3 (Homma and Woodland Hastings 1989).

Origin of Information (Naitoh, 1999a, 1999b, 2000, 2001)

The previous section showed that living systems are filled with asymmetric ratios of about 2:3. Two types of asymmetries are observed, asymmetries of size and density (frequency).

In the previous report, we clarified the reason why the size ratio of purines and pyrimidines is around silver and golden ratios (Naitoh, 1999a, 1999b, 2000, 2001).

Then, let us assume that the ancient initial molecular pool had nitrogen-containing bases having asymmetric size ratios of about 2:3 (Fig. 1). It is clarified here that the frequency ratio of purines and pyrimidines is also necessary to be about 2:3.

We will focus on the number of atoms constructing of the carbon-nitrogen rings in purines and pyrimidines. To simplify the discussion, no distinction is made between the nitrogen and carbon atoms forming the rings. The numbers of atoms in the rings of purines and pyrimidines are nine and six, respectively. It is easily understood that, from the ancient molecular pool containing carbon and nitrogen atoms, the double rings of purines with nine atoms and the single ring of pyrimidines with six atoms emerge with the frequency rate of 2:3, respectively. The law of mass conservation of atoms may offer a possible explanation for this asymmetry of frequency and density. The asymmetric size ratio easily promotes asymmetry of frequency.

Origin of Function (Naitoh, 1999a, 2000)

A possible answer is given here to the question of how the complex tRNA structure can be generated. Purines and pyrimidines are represented by "1" and "0", respectively. Then, the following string-rewriting algorithm is proposed, which is filled with the above asymmetric frequency ratio of purines and pyrimidines.
(First Step) An array with only one locus is initiated, which is either "1" or "0".
(Second Step) The "0" array in the previous generation is replaced by a new "01" array. At the same time, the "1" array in the previous generation is replaced by a new "0" array. This procedure is repeated until the array with the desired string length is generated.

It should be stressed that the one-dimensional string generated by the present algorithm is natural, because the string has a frequency of "1" and "0" in the golden ratio close to 2:3 at the infinite limit of the string length. This is the natural algorithm based on the theoretical considerations above.
(Third Step) The one-dimensional string constructed with "1" and "0" is distorted at each locus L_d in two-dimensional physical space and the loci on the right- and left- hand sides of the locus L_d face each other. When all ten loci on the right- and left- hand sides around the locus L_d are "0-0" or "1-1", the array having eleven loci, including the locus L_d, constructs a circle, that is, one of the leaves in tRNA. This is because the pairs "0-0" and "1-1", the hydrogen bonds connecting purine-purine and pyrimidine-pyrimidine, are relatively disadvantageous, although the "0-1" pair between purines and pyrimidines is widely observed in actual DNA and RNA. The locus L_d is called the key locus, when eleven loci construct a circle around the locus L_d. Next, the other loci which do not construct leaves face each other. When there are pairs of "0-1", the pairs construct trunks of tRNA (Naitoh, 1999a, 2000).

For example, we assume that the first generation is "0". After several generations based on this above algorithm, we get the following string having 89 loci,

0100101001 0010100101 0010010100 1001010010 1001001010 0101001001 0100100101 0010100100 101001001.

The string length of 89 loci is close to that of actual tRNA.

We can find that Loci 19, 35, 53, and 69 become the key loci, generating leaves in a two-dimensional structure (Fig. 2). It should be stressed that Fig. 2 is very similar to the two-dimensional clover structure of actual tRNA, although the connections in the trunks are incomplete. It is natural that the trunks of first "tRNA" was immature.

The procedure for generating the one-dimensional binary string in this algorithm proposed here is that of the DOL system, which is one of the L-systems (Prusinkiewicz and Lindenmayer 1990). This third phase is original, because the DOL system is in one-dimensional space.

The present two-dimensional clover structure can not be obtained by the production of one-dimensional binary strings using random number generators.

The natural string-rewriting algorithm filled with the 2:3 ratio promotes the first step in molecular evolution, which leads to the origin of function.

A possible generation mechanism of tRNA related to the origin of function has also been reported (Fontana and Schuster 1987). However, the model does not

explain the relation between asymmetry, information, and function. The present theory synthetically gives an explanation to the question.

The self-organizing processes of longer RNA molecules, such as mRNA and rRNA molecules may also be explained essentially by the proposed model, because longer RNA molecules can be generated by the integration of clover structures, such as tRNA molecules.

It is emphasized that the asymmetry existing in several stages of sizes and density is important for promoting the order.

Detailed Biological Auto-Catalytic Reaction Mechanism of tRNA

We again define that purine is represented by "1", while pyrimidine is "0". Then, we will propose the following natural algorithm (Naitoh, 2000).

First, let us consider the initial pool with eighty- nine nitrogenous bases, which has thirty-four purines "1" and fifty-five pyrimidines "0". (This frequency ratio is also close to 2:3.)

Thirty-four pairs of "0-1" and twenty-one bases of "0" are generated, because the pair "0-1" connected by the hydrogen bonds is relatively advantageous as seen also in the actual DNA and RNA.

Where does the remained pyrimidine "0" attach in "0-1"? We assume here that the nitrogenous base "0" can attach to each side of "0-1".

After connecting randomly, many types of strings having eighty-nine loci can be generated.

Then, the one-dimensional string constructed with "1" and "0" is distorted at each locus L_d in two-dimensional physical space and the loci on the right- and left- hand sides of the locus L_d face each other. When all ten loci on the right- and left- hand sides around the locus L_d are "0-0" or "1-1", the array having eleven loci, including the locus L_d, constructs a circle, that is, one of the leaves in tRNA. This is because the pairs "0-0" and "1-1", the hydrogen bonds connecting purine-purine and pyrimidine-pyrimidine, are relatively disadvantageous as seen in the previous sections. The locus L_d is called the key locus, when eleven loci construct a circle around the locus L_d. Next, the other loci which do not construct leaves face each other. When there are pairs of "0-1", the pairs construct trunks of tRNA.

We can find about one percent of strings among the possible strings having 89 loci take clover structure such as tRNA, as is seen in Figs. 2 and 3.

The present algorithm brings us multi-clover structures close to rRNA when the bases more than 89 are prepared in the initial pool .

It is emphasized that only the initial asymmetric frequency ratio of bases and the connections of 1-0 promote the order, that is, clover structure.

The above mechanism in one- and two- dimensional spaces can be occurred easily in the fluid-dynamic shear layers around solid walls such as rocks, that is, inhomogeneous flow fields.

The natural principle proposed above promotes the first step in molecular evolution, which leads to the origin of function.

Discussion

Let us consider Japanese poems, "Tanka" and "Haiku" constructed of the special sentences with seven and five words. This ratio is 1.40 close to silver ratio. Our brains are also comfortable in seeing objects with the silver ratio or the golden ratio, which represent B5, A4, and A3 paper sizes and the golden section in painting arts. The ratio of 2 : 3 is between the silver ratio and the golden ratio. It should be noted that the knowledge obtained here leads us to a discussion related on brains.

Here, we consider a simple genetic algorithm (SGA) having recombination, replication, and point-mutation operators and the corresponding macroscopic kinetic equation for the SGA (Naitoh 1998). The SGA uses one-dimensional binary strings. We will stress that the densities of the best-adapted string and the other strings in the SGA and also in the kinetic equation become about 3:2, when the time histories of these densities show periodic oscillation, that is, at the edge of chaos (Naitoh 1998).

Then, the macroscopic kinetic equation can also be used as an artificial intuition model, when artificial neural networks (memory) and the SGA (learning algorithm) are employed together. This is because the macroscopic kinetic equation having only four valuables can be solved with extremely small amount of calculation time. Each string in the SGA corresponds to a part of neural network. The artificial brains including this artificial intuition model, the SGA, and the artificial neural networks are filled with the 2:3 density ratios when the time-dependent densities are at the edge of chaos, that is, while the brains relax.

These may imply that the density ratio of about 2:3 is somewhere in actual brains, when the brains relax. A neural network may divide into two new different-sized neural networks, after observing the objects filled with the asymmetric ratios around silver and golden ratios.

We want to stress that several asymmetric ratios around 2:3 are possible. What do the difference between the silver and golden ratios give us? This question is still unknown.

References

Fontana, W. and Schuster, P. 1987. A computer model of evolutionary optimization. *Biophysical Chemistry* 26.

Homma, K. and Woodland Hastings, J. 1989. Cell growth kinetics. *J. of Cell Science* **92**: 303-318.

Naitoh, K. 1998. Macroscopic Kinetic Equation for a Genetic Algorithm. *Japan Journal of Industrial and Applied Mathematics* **15**: 87-133. (also *the International Institute for Advanced Studies (IIAS) Report, Kyoto*, 1998.)
Naitoh, K. 1999a. Cyto-fluid Dynamic Theory Clarifying the Origin of Biological Information and Function. *Proceedings of the Design & System Conference '99, Osaka University.* (submitted to Journal.)
Naitoh, K. 1999b. Cyto-fluid Dynamic Theory for Atomization Processes, Oil&Gas Science and Technology, Vol. 54, No.2 205-210. (Also in Proc. of COMODIA, 1998)
Naitoh, K. 2000. Mesoscopic Kinetic Theory for Fluid Phenomena and Biochemical Reaction Processes, Proc. of 3rd Japan-France Seminar of Intelligent Materials. (to be published in J. of Adv. Sci.)
Naitoh, K. 2001. Cyto-fluid Dynamic Theory, Japan Journal of Industrial and Applied Mathematics, Vol.18, No.1.
Prusinkiewicz, P. and Lindenmayer, A. 1990. *The Algorithmic Beauty of Plants*: Springer-Verlag.
Watson, J. D., Hopkins, N. H., Roberts, J. W., Steitz, J. A.,and Weiner, A. M. 1987. *Molecular Biology of the Gene*, Fourth edition: The Benjamin/Cummings Publishing Company, New Jersey, U.S.A.

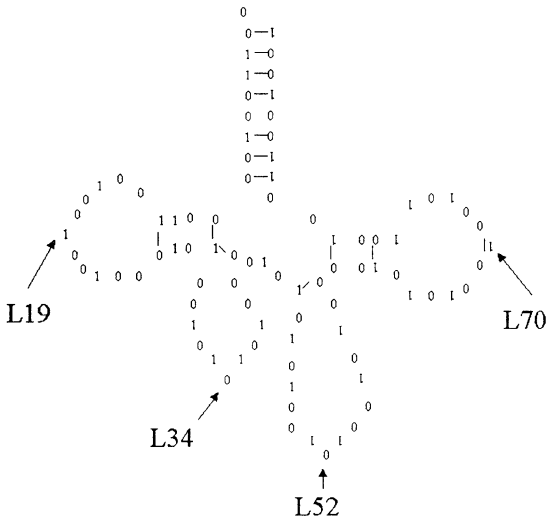


Fig. 3 Two-dimensional clover structure of tRNA obtained by the present detailed algorithm.

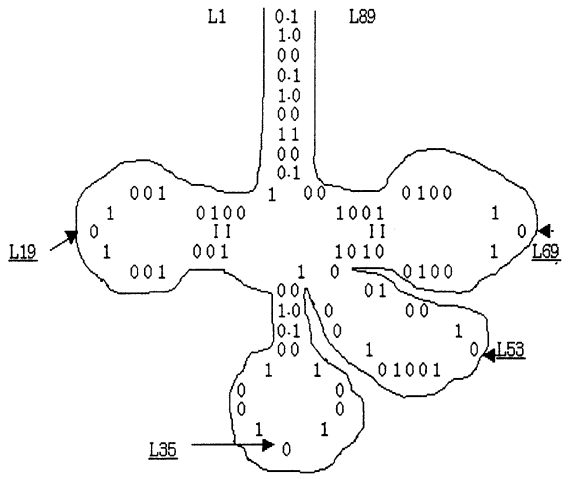


Fig. 2 Two-dimensional clover structure of tRNA obtained by the DOL algorithm.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Complex Organics in Space to Life: A New Scenario of Chemical Evolution

°Kensei Kobayashi[1], and Hiroshi Yanagawa[2]

[1]Department of Chemistry and Biotechnology, Faculty of Engineering, Yokohama National University, Hodogaya-ku, Yokohama 240-8501, Japan
E-mail: kkensei@pchem.bsk.ynu.ac.jp
[2]Department of Applied Chemistry, Faculty of Science and Technology, Keio University,
3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan
E-mail: hyana@applc.keio.ac.jp

Abstract. Before the first terrestrial organism was generated on the primitive Earth about four billion years ago, bioorganic compounds should have been supplied to the primitive ocean. How were bioorganic compounds formed and organized into the first "life"? A conventional hypothesis tells that amino acids, nucleic acid bases and sugars formed in the ocean were condenced together to form proto-proteins and proto-RNAs. Recent analyses of extraterrestrial organic compounds and products of laboratory experiments simulating primitive environments suggest that such "clean chemistry" hardly occured abiotically. Here we introduce a new scenario of chemical evolution via complex organic compounds toward the first "life".

Keywords. Extraterrestrial organic compounds, origins of life, chemical evolution, amino acids, cosmic radiation, complex organics.

1. INTRODUCTION

The life's origin is one of the most fundamental questions remained for the humankind. In 1920s, the idea of the chemical evolution hypothesis of origin of life was first proposed by Oparin and Haldane. Since Miller's historical spark discharge experiment [1], a wide variety of laboratory experiments have been conducted to prove this hypothesis.

In the earlier studies of chemical evolution, the following scheme was widely accepted: (i) Active comounds such as hydrogen cyanide and formaldehyde were abiotically produced from strongly reduced terrestrial atmosphere by thundering, solar UV radiation, and so on. (ii) Reactions among these active compounds gave bioorganic monomers like amino acids, nucleic acid bases and sugars. (iii) These monomers polymerized together to give polypeptides and polynucleotides. (iv) Finally "life" was generated, based on these biologically-important polymers.

Recent findings and theories on the history of earth and other planets gave new insights to the chemical evolution and the origin of life. For example, the discovery of submarine hydrothermal vents in the late 1970's [2] has drastically changed the image of the primordial sea where life was believed to be born. Here, we would like to introduce the latest idea of origin of life, centering around the formation of complex organic compounds in primitive terrestrial and extraterrestrial environments.

2. ENERGETICS OF BIOORGANIC COMPOUNDS IN PRIMITIVE EARTH ATMOSPHERE

Urey [3] postulated that the primitive Earth atmosphere was strongly reduced. Miller [1] reported that bioorganic monomers such as amino acids were easily formed in a mixture of methane, ammonia, hydrogen and water ("a strongly reduced agas mixture") by spark discharges. Other energy sources such as ultraviolet light and heat were also applied to the same kind of gas mixtures, and amino acids were successfully found in the products.

In these days, the primitive Earth atmosphere was discussed in the context of newly-developed planetary sciences. It has been suggested that high velocity impacts of planetesimals onto a growing planet can result in the impact-degassing of volatiles and the formation of an impact-induced atmosphere at high temperature[4]. According to these hypotheses, the resulting impact-induced atmosphere contains carbon monoxide or carbon dioxide as a major carbon source [5]. No or only a trace of amino acids was obtained in simulated primitive atmosphere experiments when carbon monoxide or carbon dioxide is used as a carbon source [6][7].

Kobayashi et al.[8] examined possible roles of cosmic rays in abiotic formation of bioorganic compounds. A mixture of carbon monoxide, carbon dioxide, nitrogen and water was irradiated with high energy (3-40 MeV) protons generated a van de Graaff accelerator or an SF cyclotron. Fig. 1 shows a chromatogram of the product before and after acid hydrolysis. It was proved that a number of "precursors" of amino acids such as glycine and alanine were abiotically produced from the gas mixture in high yield. The G-value (number of formed molecules per 100 eV) of glycine was as large as 0.02.

It was suggested that cosmic rays were the most effective energy sources among all the energy sources available on the primitive Earth [8].

3. COMPLEX ORGANIC COMPOUNDS IN SPACE

It is known that a wide variety of organic compounds are present in extraterrestrial environments. For example, infrared spectrometers on board Voyger 1 and 2 prevailed the presence of organic compounds in such solar system bodies as Jupiter and Titan (a Moon of Saturn) [9].

Among these extraterrestrial organic compounds, cometary organics are most important since they could be sources for the first terrestrial biosphere after their collisions onto the primitive Earth [10].

Mass spectrometers on board Vega 1 and Giotto showed that cometary dusts contained a wide variety of complex organic compounds [11]. It has been hypothized that these cometary organics were formed in ice mantles of interstellar dust particles (ISD) in dark clouds by cosmic rays and UV light. Kasamatsu et al. [12] showed that icy mixture of carbon monoxide, ammonia and water ("simulated ISD ice mantles") gave amino acid precursors.

4. SUBMARINE HYDROTHERMAL SYSTEMS AS POSSIBLE SITES OF ORIGINS OF LIFE

It is expected that organic compounds formed either in primitive earth atmosphere or in interstellar media were collected in primitive sea, where chemical evolution toward the origin of life proceeded.

In the late 1970s, very hot water over 300°C was

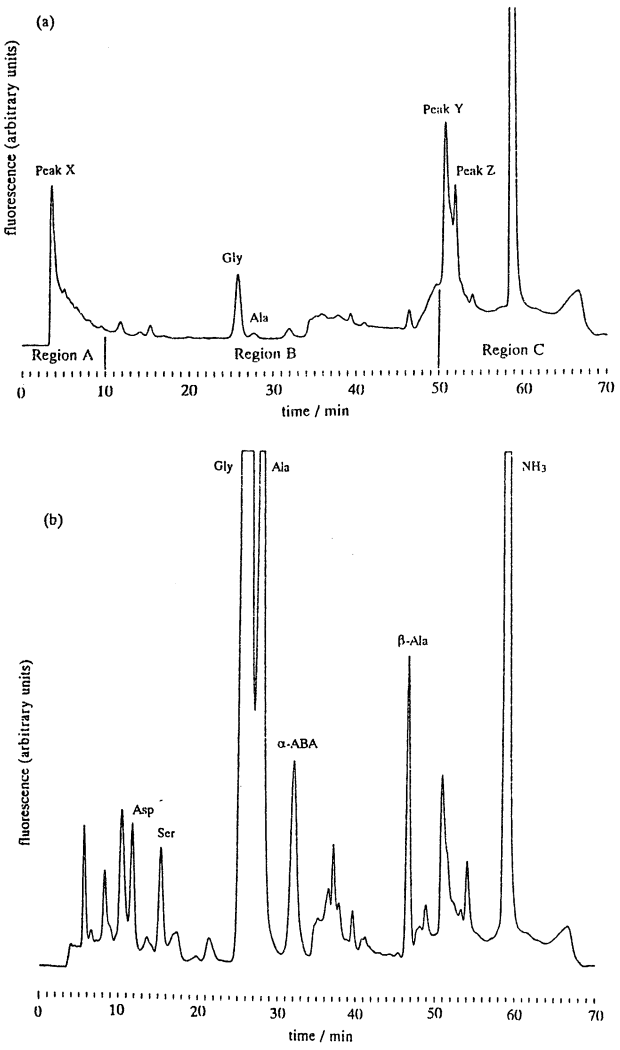


Fig. 1. Cation-exchange chromatograms of a proton irradiation product from a mixture of carbon monoxide, nitrogen and water. (a) Analyzed before cid-hydrolysis; (b) analyzed after hydrolyzed with 6 M HCl at 110°C for 24 h.

found venting at the Galapagos spreadomg center [2]. Such places were named "submarine hydrothermal vents". Submarine hydrothermal vents have been found at various sites along plate boundaries all over the world. Several extremely thermophilic organisms with optimum growth temperatures at 100°C or above have been found near submarine hydrothermal vents [13]. Extreme thermophiles are believed to be located very close to the root of the phylogenetic tree. Therefore, the last universal cellular ancestor (LUCA) or the "commonote" are suggested to have been an extreme thermophile, suggests that submarine hydrothermal systems were probable sites of the origin of life on the Earth [14].

Submarine hydrothermal vent systems have the following interesting characteristics: (i) Thermal energy

simple starting materials; (ii) they represent reducing conditions, which are favorable for organic syntheses; (iii) The thermal fluid contains a quite high concentration of transition metal ions that catalyze chemical and biochemical reactions.

Yanagawa and Kobayashi [15] did experiments simulating submarine hydrothermal vent environments. They found that amino acid precursors formed when aqueous solution simulating hydrothermal vent medium was heated at 325°C for 1.5-12 hours under pressuruzed with a gas mixture of methane and nitrogen. Yanagawa et al. [16] also found that many microspheres were formed when an aqueous solution containing glycine, L-alanine, L-valine, and L-aspartic acid was put into a glass tube and heated at 250°C and 134 atm for 6 h.

In our experiments, "modified hydrothermal vent medium" was pressurized and heated in a closed vessel. The actual hydrothermal system is, however, considered as "a flow system" where quenching of the heated fluid by cold seawater occurs. Further simulation experiments using the flow system (Fig. 2) will help clarify the extent to which the present results reflect what can happen in natural system [17].

5. A NEW SCENARIO OF CHEMICAL EVOLUTION TOWARD LIFE

The conventional scenario of chemical evolution is as follows: (1) Active intermediates such as HCN and HCHO were formed in primitive atmosphere by spark discharge or other energy sources; (2) amino acids, nucleic acid bases and sugars were formed in aqueous solution from the intermediate; (3) amino acids were condensed togethere to form oligopeptides, and bases, sugars and phosphates gave oligonucleosides; (4) Oligopeptides or oligonucleosides with catalytic activities were selected as proto-enzymes.

In our laboratory simulation, "complex organic compounds", whose molecular weights were ca. 1000, were fomed from a gas mixture of carbon monoxide, nitrogen and water by irradiation with high energy particles [18]. This complex organic compounds gave amino acids and uracil [19] when they were dissolved in water, and show some catalytic acitivities including esterase activity. These results did not agree with the conventional scenario.

As described previously, organic compounds found in extraterrestrial bodies like comets and meteorites are also complex organics. Free amino acids are not stable in

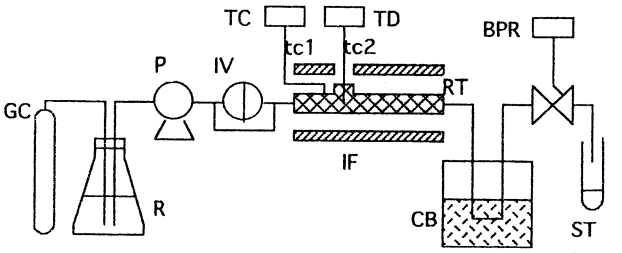


Fig. 2. Supercritical water flow reactor simulating submarine hydrothermal systems.
GC: gas cyllinder, R: carrier reservoir, P: HPLC pump, IV: injection vulve, IF: infrared gold image furnace, RT: reaction tube, TC: temperature controller, TD: temperature display, tc1 and tc2: thermocouples, BPR: back-pressure regulator, CB: cold bath, ST: sampling tube.

terrestrial and extraterrestrial environments: Possible presence of glycine in molecular clouds were reported [20], but it has not confirmed; Kissel and Krueger [11] reported that a wide variety of organic compounds were present in cometary dusts, but with exception of amino acids. These facts suggest that "free amino acids" are not common compounds in extraterrestrial environments, while amino acid precursors seem to be quite popular in space.

Here we propose a new scenario of chemical evolution as follows:
(1) Complex organic compounds were formed in planetary atmosphere or in interstellar media from simple molecules like carbon monoxide and nitrogen by high energy particles.
(2) Complex organics dissolved in primeval sea gave catalytic molecules togethere with components of proteins and RNAs (e.g., amino acids and nucleic acid bases).
(3) Autocatalytic molecules were selected from abiotic catalysts to form a self-multiplication system.
(4) The self-multiplication system refined itself by the incorporation of the components of proteins or nucleic acids to evolve into "a protein world" or "an RNA world".

It is quite important to detect autocatalytic organic molecules in a "primordial soup" to test the new chemical evolution scenario described above.

Acknowledgements

The present work was partly supported by a Grant-In-Aid (No. 11640488) from the Ninistry of Education, Science nd Culture, Japan.

References

[1] Miller SL (1953), A production of amino acids under possible primitive Earth conditions. Science 117, 528-529.

[2] Corliss JB, Dymond J, Gordon LI, et al. (1979), Submarine thermal springs on the Galapagos Rift. Science 203: 1073-1083.

[3] Uery, HC (1952), "The Planets, Their Origin and Development," Yale University Press, New Haven.

[4] Matsui T, Abe Y (1986), Impact-induced atmospheres and ocean on Earth and Venus. Nature 322: 526-528.

[5] Kasting J (1990), Bolide impacts and the oxidation state of carbon in the Earth's early atmosphere. Origins Life Evol. Biosphere 20:199-231.

[6] Schlesinger G, Miller SL (1983), Prebiotic synthesis in atmospheres containing methane, carbon monoxide and carbon dioxide. J. Mol. Evol. 19:376-390.

[7] Bar-Nun A, Chang S (1983), Photochemical reactions of water and carbon monoxide in Earth's early atmosphere. J. Geophys. Res. 88:6662-6672.

[8] Kobayashi K, Kaneko T, Saito T, Oshima T (1998), Amino acid formation in gas mixtures by high energy particle irradiation. Origins Life Evol. Biosphere 28: 155-165.

[9] Carle G, Schwartz D, Huntington J (Eds.) (1992), Exobiology in Solar System Exploration, NASA SP 512.

[10] Chyba C., Sagan C (1992), Endogenous production, exogenous delivery and impact shock snthesis of organic molecules: an inventory for the origins of life. Nature 355:125-132.

[11] Kissel J, Krueger FR (1987), The organic componemt in dust from comet Halley as measured by the PUMA mass spectrometeron board Vega 1. Nature 326: 755-760.

[12] Kasamatsu T, Kaneko T., Saito T, Kobayashi K (1997), Formation of organic compounds in simulated interstellar media with high energy particles. Bull. Chem. Soc. Jpn. 70:1021-1026.

[13] Fiala G, Stetter KO (1986), Pyrococcus furiosus sp. Nov. Represents a novel genus of marine heterotrophic archae bacteria growing optimally at 100°C. Arch. Microbiol. 145:56-61.

[14] Holm NG (Ed.) (1992), Marine Hydrothermal Systems and the Origin of Life, Kluwer,Dordrecht, 1992.

[15] Yanagawa H, Kobayashi K (1992), An experimental approach to chemical evolution in submarine hydrothermal systems. Origins Life Evol. Bioshere 22:147-159.

[16] Yanagawa H, Kojima K (1985), Thermophilic microspheres of peptide-like polymers and silicates formed at 250°C. J. Biochem. 97:1521-1524.

[17] Islam MN, Kaneko T, Kobayashi K (2000), Construction of a supercritical water flow reactor to simulate the submarine hydrothermal system. Viva Origino 28:9.

[18] Kobayashi K, Sato T, Kajishima S, Kaneko T, Ishikawa Y, Saito T (1997), Possible complex organic compounds on Mars. Adv. Space Res. 19:1067-1076.

[19] Kobayashi K, Tsuji T (1997), Abiotic synthesis of uracil from carbon monoxide, nitrogen and water. Chem. Lett. 1997:903-904.

[20] Snyder LE (1997), The search for interstellar glycine. Origins Life Evol. Biosphere 27:115-133.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Fitness landscape of biopolymers and efficient optimization strategy in evolutionary molecular engineering.

Takuyo Aita[1,2], Yuzuru Husimi[2]

[1]Tsukuba Research Institute, Novartis Pharma K. K., Ohkubo 8,Tsukuba, 300-2611, Japan,
[2]Department of Functional Materials Science, Saitama University, Urawa 338-8570, Japan
aita@evolve.fms.saitama-u.ac.jp; husimi@fms.saitama-u.ac.jp

Abstract

Evolution of biopolymers, such as proteins or functional RNA and DNA, is considered as "adaptive walk" or "hill climbing" process on "fitness landscapes" in sequence space, where fitness is defined as a measure of catalytic activity, affinity or stability of biopolymers. In evolutionary molecular engineering, we should take an efficient adaptive walk strategy based on the knowledge for the statistical feature of fitness landscapes of biopolymers. Regarding "local fitness landscapes" around native proteins, we have interpreted several examples of real landscapes by analyzing fitness data of various mutants for prolyl endopeptidase, thermolysin and other proteins. As a result, we found that the rough mutational additivity holds in biopolymer's fitness and the local fitness landscapes are of the rough Mt. Fuji-type. This result leads to an efficient evolution strategy through simultaneous accumulation of advantageous mutations. This strategy is regarded as an "adaptive leap" strategy. Regarding "global fitness landscapes", a hypothesis of "neutral network landscape" is prevailing, where a neutral network (graph) is formed with various sequences folding into a same structure and is webbing on global sequence space.

key words: evolutionary molecular engineering, evolution strategy,molecular evolution, fitness landscape, sequence space, adaptive walk

1 Introduction

While evolutionary molecular engineering has the potential to create useful functional biopolymers, it also provides the opportunity to uncover the principles of the emergence of functional biopolymers or molecular evolution. One of the most important issues among them may be "fitness landscapes" of biopolymers. In evolutionary molecular engineering, the concept of "fitness" in biology is expanded to a quantitative measure of a certain physicochemical property of a biopolymer (i.e. enzymatic activity, affinity to a ligand or structural stability), and the Darwinian evolution of a biopolymer is considered as an adaptive walk on a fitness landscape [1,2,3] . It is important to know the statistical properties of real landscapes, that is the "general shapes" of various landscapes, the "mean slope" of a landscape, and so on.

Almost all results of mutagenesis experiments ever performed demonstrate that real local fitness landscapes for current proteins are almost compatible with a model of the Mt. Fuji-type landscape based on mutational additivity of biopolymers [4]. We consider this landscape as the first approximation of a real landscape. It can be regarded as an "ideal landscape" analogous to an "ideal gas". For the purpose of exploring fitness landscapes, we have studied a method to analyze mutant fitness data and to interpret the shape of the landscapes by using our model of the Mt. Fuji-type fitness landscape. In addition to this, we proposed an efficient evolution strategy regarded as an adaptive leap on a landscape.

2 Analysis of a local fitness landscape: application to thermostability landscape of prolyl endopeptidase

Uchiyama *et al.* have tried to improve the thermostability of prolyl endopeptidase (PEP, EC 3.4.21.26) from

Flavobacterium meningosepticum [5] , using the adaptive walk technique of evolutionary molecular engineering. The thermostability was evaluated by measuring the half-life $\tau_{1/2}$ of deactivation at 60°C. As a result, they identified 8 amino acid substitutions improving the thermostability, E67Q,F70L,A148T,N387K,S475G,G652V,S653A and Q656R. Subsequently, they prepared 19 multiple mutants by recombining 8 mutants, and evaluated the thermostability of each clone. We here define the fitness of each clone as the natural logarithm of the half-life and analyzed the mutant fitness data based on the following fitness function:

$$F_{\mathrm{P}} = F_{\mathrm{O}} + \sum_{j=1}^{\nu} w_j(\alpha_{\mathrm{P}j})$$

$$w_j(\alpha) \begin{cases} = 0, & \text{if } \alpha = \alpha_{\mathrm{O}j} \\ < 0, & \text{if } \alpha \neq \alpha_{\mathrm{O}j} \end{cases}$$

F_{P} is the fitness of a sequence P, and O represents the optimum sequence that corresponds to the single peak on the model landscape. ν represents the number of all mutated sites and j represents an arbitrary site among them. $\alpha_{\mathrm{P}j}$ represents the particular residue at the jth site in the given sequence P. $w_j(\alpha)$ is the "site-fitness", that is a contribution from a certain residue α at the jth site. We determined the parameters F_{O} and $w_j(\alpha)$'s through parameter fitting. The smoothness of the landscape is evaluated with the ratio θ of the mean slope to the degree of roughness:

$$\theta = \frac{1/\nu \sum_{j=1}^{\nu} \sum_{\alpha} |w_j(\alpha)|}{\sqrt{1/n \sum_{\mathrm{i}=1}^{n} (\mathcal{F}_{\mathrm{P_i}} - F_{\mathrm{P_i}})^2}},$$

where $\mathcal{F}_{\mathrm{P_i}}$ and $F_{\mathrm{P_i}}$ are the experimental fitness and theoretical one of a sequence $\mathrm{P_i}$, respectively. n is the number of fitness data available in the analysis. As the θ-value becomes larger, the surface becomes smoother in comparison with the mean slope.

Fig.1 and Fig.2 show the mutational effect for the four component mutations: E67Q, F70L, A148T, G652V. The 16 mutants taken into analysis are as follows: the wild-type, single mutants E67Q, F70L, A148T, G652V, double mutants (E67Q, F70L), (E67Q, A148T), (E67Q, G652V), (F70L, A148T), (F70L, G652V), (A148T, G652V), triple mutant (E67Q, F70L, A148T), (E67Q, F70L, G652V), (E67Q, A148T, G652V), (F70L, A148T, G652V), quadruple mutants (E67Q, F70L, A148T, G652V). $\nu = 4$ and $n = 16$ in this case.

As a result, the local fitness landscape is very similar to the Mt. Fuji-type landscape having the mean slope of

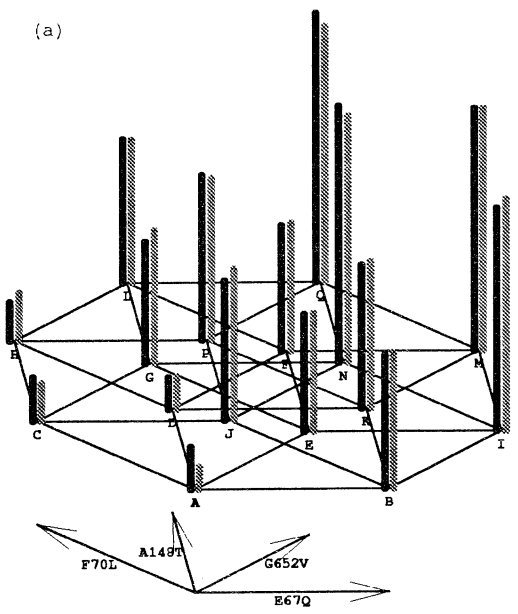


Fig.1. A local thermostability landscape for prolyl endopeptidase The sequence space is projected on the plane of the page. The sequences denoted by letters are located at nodes: "A" denotes the wild-type and "B","C",$\cdots$,"Q" denote mutants. The fitness of a sequence is represented by a bar standing at near the corresponding letter (arbitrary unit). The black bars and gray bars are experimental fitnesses and theoretical ones obtained from the analysis, respectively. Adapted from [4].

$\langle |w_j(\alpha)| \rangle = 1.3$ and $\theta = 6.9$. In other words, the mutational effect for the component mutations are almost additive. In addition to this, we have analyzed various mutant fitness data for other proteins [4] and confirmed the hypothesis that mutational effects of current biopolymers are almost additive.

3 Mutation scrambling method: an efficient evolution strategy through simultaneous accumulation of advantageous mutations

Efficient optimization methods in evolutionary molecular engineering for such a Mt. Fuji-type landscape seem to be based on simultaneous accumulation of advantageous single point mutations identified in a preliminary selection from a random single point mutant pool [6,7] . The rough additivity suggests that the sequence with all of the advantageous point mutations is not necessarily the sequence with the highest fitness. Efficient optimization methods should find the best combination of the mutant residues and wild-type residues. One of the methods is the "mu-

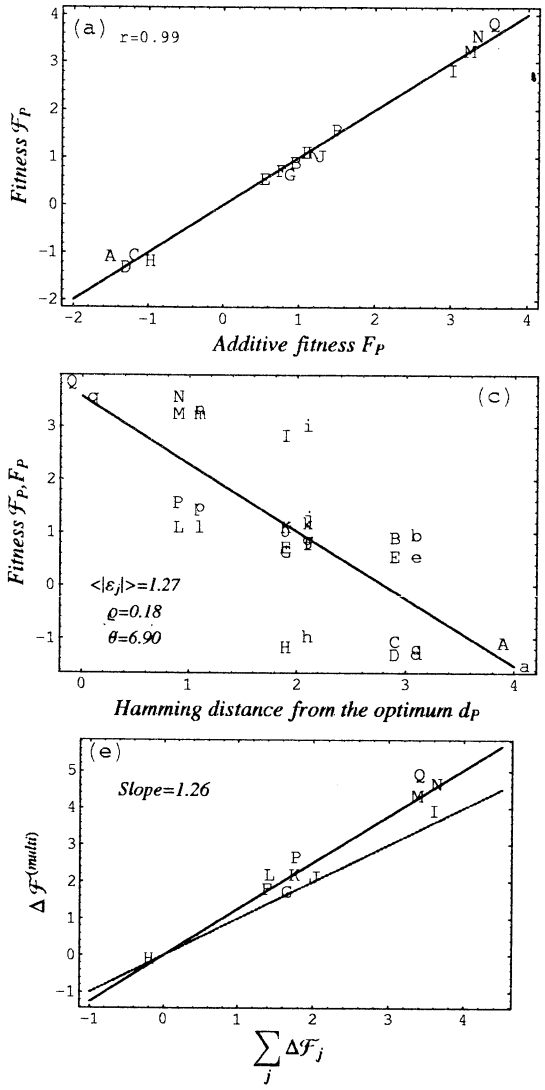


Fig.2. Results of analysis of the local thermostability landscape.
(a) Correlation between experimental fitnesses $\mathcal{F}_P$ (ordinate) and theoretical ones F_P (abscissa). (c) A sketch of each landscape. The abscissa is the Hamming distance from the optimum "Q" to each sequence. The ordinate is experimental fitnesses (upper case letter) or theoretical ones (lower case letter). The solid line represents the mean slope of the landscape. (e) The ordinate is the changes in fitnesses for the multiple mutant ($\Delta\mathcal{F}^{(\mathrm{multi})}$) and the abscissa is the sum for the component mutants ($\sum_j \Delta\mathcal{F}_j$). The original sequence is the wild-type "A". The regression line (solid line) has a slope of 1.26 with correlation coefficient of 0.97. The dashed line has a slope of 1. Adapted from [4].

tation scrambling" method [8]. Fig.3 shows the scheme of the mutation scrambling method, which consists of the following processes. (1) Several advantageous single point mutations are identified. (2) Mutant fragments and wild-type fragments are prepared by PCR amplification with appropriate primers. (3) Various multiple point mutants incorporating the advantageous mutant residue or wild-type residue at each of the selected sites are generated by overlapping elongation by PCR without primers. The fittest is selected from the mutant pool.

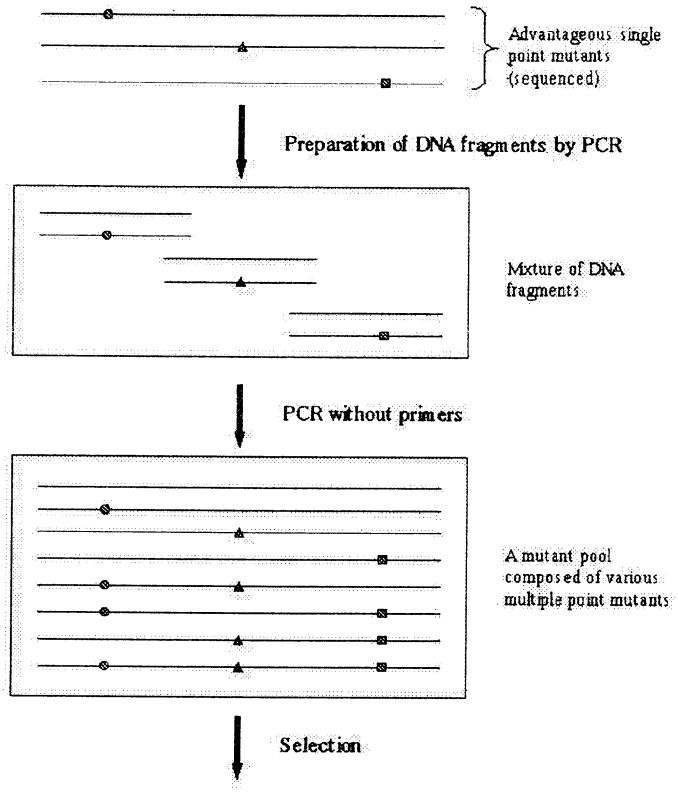


Fig.3. Scheme of the mutation scrambling method [8]. The circle, triangle and square stand for advantageous single point mutations. The details of the method are in the text.

The fitness distribution in the mutant pool is controlled by the mixing ratio of the mutant residues to the wild-type residues [7] . Let r be the mixing fraction of each mutant fragment. That is, the mixing ratio (mutant fragment/wild-type fragment) is given by $r/(1-r)$. Therefore, in the mutant pool resulting from the mutation scrambling, the frequency of d-fold point mutants obeys a binomial distribution $\binom{\nu}{d} r^d (1-r)^{(\nu-d)}$. Let ΔF be a variable representing the change in fitness from the wild-type P to an arbitrary multiple point mutant. As a result deduced from the previous hypothesis, the probability density of ΔF with a given r is approximately described as

follows:

$$\mathcal{N}(\Delta F | r \sum_{j=1}^{\nu} \Delta w_j, r(1-r) \sum_{j=1}^{\nu} \Delta w_j{}^2),$$

where $\mathcal{N}(x|E,V)$ denotes the Gaussian distribution of a variable x with mean E and variance V. Δw_j is the change in the additive fitness and the sum is taken over ν component sites of interest. Fig.4 shows the probability density of ΔF (=mutant fitness distributions) with various r-values. We can see that the center of the mutant fitness distribution linearly shifts up to the maximum value, as the r-value increases. The evolution strategy with high r-values raises the frequency of superior multiple point mutants having high fitnesses. However, the strategy with extremely high r-value reduces the frequency of the superior mutants. This reduction stems from non-additive effect resulting from inter-residue interaction. Therefore, there is an optimal r-value $r_{\rm opt}$ giving the maximum frequency of the superior mutants. We estimated $r_{\rm opt} = 0.7 \sim 0.9$ for real biopolymers.

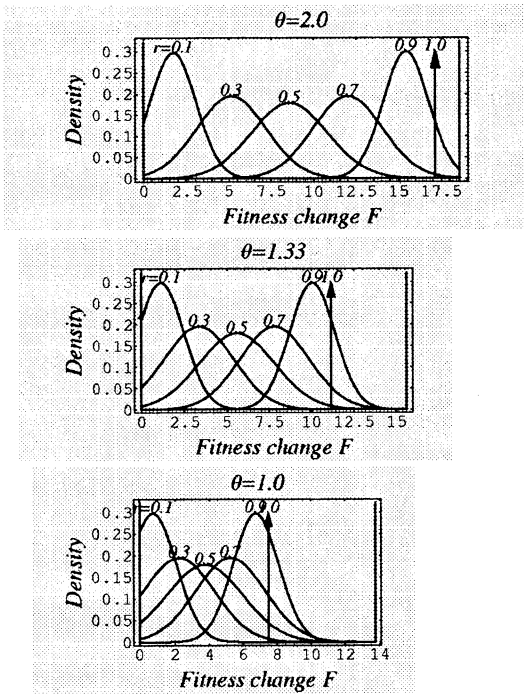


Fig.4. Mutant fitness distributions with various mixing fractions r. The abscissa is the fitness change ΔF from the wild-type fitness. The ordinate is the probability density. The rightmost bar represents the fitness change for the fittest in the mutant pool in all cases except $r = 0$ or $r = 1$. $r = 0.1$, 0.3, 0.5, 0.7, 0.9, 1.0. $\nu = 30$.

4 Conclusion

Local fitness landscapes for several proteins are rough Mt.Fuji-type based on the rough mutational additivity. Following this result, we proposed an efficient evolution strategy using mutation scrambling method with biased mixing ratio of the mutant residues to the wild-type residues. The fitness distribution in the mutant pool is controlled by the mixing ratio. We expected the optimal mixing ratio between 7/3 and 9/1 which efficiently generates superior multiple point mutants with high fitnesses. Particularly, this strategy works well in cases where the number of component mutations is large and the size of the population to be screened is small.

5 Acknowledgments

This work was performed as a part of the R&D Project of the Industrial Science and Technology Frontier Program supported by NEDO (New Energy and Industrial Technology Development Organization).

References

[1] Maynard-Smith, J. (1970). *Nature* **225**, 563-564.

[2] Eigen, M. (1985). *Ber. Bunsenges. Phys. Chem.* **89**, 658-667.

[3] Kauffman, S.A. (1993). *The Origin of Order.* Oxford: Oxford University Press.

[4] Aita, T., Uchiyama, H., Inaoka, T., Nakajima, M., Kokubo, T. & Husimi, Y. (2000). Analysis of local fitness landscape with a model of the rough Mt. Fuji-type landscape: application to prolyl endopeptidase and thermolysin. *Biopolymers* **54**, 64-79.

[5] Uchiyama, H., Inaoka, T., Ohkuma-Soyejima, T., Togame, H., Shibanaka, Y., Yoshimoto, T and Kokubo, T. (2000). Directed Evolution to Improve Thermostability of Prolyl Endopeptidase. *J. Biochem.* **128**, 441-447.

[6] Kuchner, O. and Arnold, F.H. (1997). Directed evolution of enzyme catalysts. *TIBTECH* **15**, 523-530.

[7] Aita, T. & Husimi, Y. (2000). Theory of an Evolutionary Molecular Engineering through Simultaneous Accumulation of Advantageous Mutations. *J.theor.Biol.* **207**, 543-556.

[8] Uchiyama, H. (1999). Mutation scrambling for *In Vitro* evolution to improve thermostability of prolyl endopeptidase (in Japanese). *BIO INDUSTRY* **16**, 27-35.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Novel Non-Metric MDS Algorithm with Confidence Level Test

oY-h. Taguchi
Department of Physics
Chuo University
Kasuga, Bunkyo-ku, Tokyo 112-8551
Japan, e-mail:tag@granular.com

Y. Oono
Loomis Laboratory of Physics
University of Illinois
at Urbana-Champaign
1110 W Green, Urbana, IL 61801, USA.

Abstract

A novel algorithm for non-metric multidimensional scaling (NMDS) method is proposed that is closely related to a means to evaluate nonparametric confidence levels of resultant configurations. Our algorithm is efficient enough to allow analyses of relations among much more than 1000 items with an inexpensive personal computer. The algorithm is illustrated with preliminary applications to, e.g., DNA sequence data.

Key Words: non-metric multidimensional scaling, nonparametric confidence level test

1 Introduction

Increasing importance of bioinformatics certainly requires efficient methods to handle very large complicated data sets to extract patterns and rules. To this end in mind in this paper we report a new efficient algorithm for non-metric multidimensional scaling (NMDS). Multidimensional scaling (MDS) (see, e.g., Ref.[1]) is a major branch of multivariate analysis chiefly used in social sciences, psychology, and occasionally in ecology to visualize hidden relations among objects of interest (henceforth we call them operational taxonomic units (OTU)). Its essence is to find, in a certain metric space, a configuration (constellation) of the points corresponding to OTUs that is compatible with the given dissimilarity relations among them. The resultant configuration often visually exhibits the relations/structures hidden in the original data.

The metric MDS attempts to construct a configuration of OTUs in which the dissimilarity measure $\delta(i,j)$ for each OTU pair (i,j) is proportional to the distance $d(i,j)$ in the constructed configuration. The NMDS [2] attempts to construct a configuration of OTUs whose distances $d(i,j)$ preserve the rank ordering of the dissimilarities in the original data. That is, for any two pairs of OTUs, (i,j) and (k,l), the constructed configuration satisfies $d(i,j) > d(k,l)$ if $\delta(i,j) > \delta(k,l)$. NMDS can be applied to the rank data of dissimilarities (that is, qualitative data), and can often recover the metric MDS results when there are sufficiently many OTUs. Therefore, NMDS is one of the most versatile multivariate analysis methods. Usually, NMDS can deal with much more robust data than MDS can handle. Thus, in this paper, we construct a "pure" NMDS, maximally eliminating metric elements (even the so-called $\hat{d}$) in contrast to the conventional NMDS.

Our novel algorithm is closely related to the optimization of the measure of nonparametric statistical confidence level of the resultant configuration. Besides, the algorithm is very efficient so that it even enables us to analyze $N = 10^4$ OTUs (this is not the number of binary relations that is a much larger number $\sim N^2/2$) with an inexpensive desktop machine. A 1000 OTU example can be treated successfully. Since our algorithm is of order $N^2 \ln N$ for updating the configuration (that is, essentially the number of the relations we handle, so greater efficiency is hardly possible), it is clear that much larger number of OTUs can be practically analyzed with NMDS.

The paper is organized as follows. In Section 2, the conventional NMDS methods are reviewed to show the presence of additional and unnecessary constraints. Section 3 discusses what "pure NMDS" should be, and our new NMDS algorithm is proposed in Section 4. A method to measure the nonparametric statistical confidence level of the resultant configurations is proposed in Section 5. Finally, we exhibit an application of NMDS to a molecular biological example in Section 6. Section 7 is a summary.

2 Conventional NMDS

Suppose an (increasing) ordering of the pair dissimilarities of N points (OTUs) $\{1, 2, \cdots, N\}$ is given. That is, for any two pairs of points (i, j) and (k, l) such that $i, j, k, l \in \{1, 2, \cdots, N\}$ we assume that we can tell which dissimilarity of the two pairs is not smaller than the other. A typical NMDS problem may be summarized as follows: In a given metric space $\mathcal{R}$ find a configuration of N points the (increasing) order of whose pair distances is as close as possible to that of the given pair dissimilarities

All the conventional NMDS methods assume a certain intermediate pair distance (e.g., $\hat{d}$ below) that is compatible with the actual ordering of the pair dissimilarities and is simultaneously as compatible as possible with the pair distances d in the estimated configuration of the OTUs. There are many varieties of the conventional approach.

Kruskal's approach may be summarized as follows. We start with a set of pair distance values d that are computed from a trial N-point configuration in $\mathcal{R}$. The greatest convex minorant (that is, the lower boundary of the convex hull) of the plot of d against the ranking of the distances in the original data defines $\hat{d}$. d is then revised to minimize the stress: a suitably normalized discrepancy between d and $\hat{d}$. After this, the whole procedure is iterated until the stress becomes tolerably small. There are several implementations of this idea, but we do not discuss this approach any further (See, for example, [1] and [3]).

Guttmann [3] constructed d^*, which is the rearrangement of d according to the actual ordering of the given dissimilarities δ. Then, the stress $S_G = ||d - d^*||$ is minimized for d. He showed that this procedure and Kruskal's procedure are equivalent. In the actual calculation, d^* is not directly considered as a functional of d, but is treated as given just as $\hat{d}$ in each updating process. Then, the procedure is iterative as in the methods in the preceding paragraph.

As has been seen above, the essence of the conventional NMDS is to choose $\hat{d}$ as close to d as possible under the condition that it is monotone with respect to the actual ranking of the dissimilarities in the original data. Depending on the interpretation of "as close as," different methods have been proposed as summarized on p43 in Ref. [4]. KYST is the program based on the least square fit. Spline function fitting, etc., are conceivable and actually used. On p52 - 58 in Ref. [4] the letter 'R' is presented with about 30 representative points, and NMDS is performed with various algorithms. For example, the results of MDSCAL and KYST are <u>different</u> (KYST does not affinely skew 'R' as MDSCAL does). That is, the choice of $\hat{d}$ affects the outcome.

Thus it is clear that the conventional NMDS introduces extra constraints. If we wish to be faithful to the basic idea of NMDS according to its creator (Shepherd), it is an inevitable conclusion that $\hat{d}$ is required only by technical reasons for implementation. The input data is the ranking of δ. Then, what we must compare with it is the ranking of d. If the data set is purely without metric, but only with rank ordering, then the actual size of d should not matter (should not affect the outcome[1]). We must note that all the procedures to determine $\hat{d}$ (or d^*) so far proposed are more or less affected by the actual values (or their ratios) of d.

3 Pure NMDS

To implement NMDS in a way faithful to the idea of its creator, we must not introduce any extra information, bias, structure, etc., into the given information: the ranking of dissimilarities between pairs. We do not have any information about the actual pair distances. We wish to look for configurations of N points in $\mathcal{R}$ whose pair distances have the ranking maximally compatible with the given dissimilarity ordering.

Logically, there can be two approaches:
(A) We compare the distances d obtained from the configuration in $\mathcal{R}$ and the given ordering of the pair dissimilarities δ through their ranks only.
(B) We compare the distances d obtained from the configuration in $\mathcal{R}$ and all the possible distances that are compatible with the pair dissimilarity ranking.

The method (B) was recently proposed and implemented by Trosset [5].

The approach (B) may be interpreted as an unbiased encoding of the rank ordering into the actual distances. In order not to bias the information due to encoding schemes, (B) takes into account of the totality of possible encoding schemes consistent with the rank ordering in the original data. Thus, logically (A) and (B) are equivalent, but since encoding into distances is an extra step, (B) is conceptually and practically less direct than (A).

[1] if it does, it implies that the result of the NMDS is not unique.

4 New NMDS Algorithm

In the following, we propose a new algorithm for NMDS that realizes approach (A).

The basic idea of the algorithm is as follows: in a metric space $\mathcal{R}$ (in this paper, for simplicity, we assume this to be a D-dimensional Euclidean space $\mathcal{R} = \boldsymbol{R}^D$) N points representing the OTUs are placed as an initial configuration. For this initial trial configuration we compute the pair distances $d(i,j)$, and then rank them according to their magnitudes. Comparing this ranking and that according to the dissimilarity $\delta(i,j)$, we compute the 'force' that moves the points in $\mathcal{R}$ to reduce the discrepancies between these two rankings. The 'force' along the line connecting OTUs i and j is chosen to decrease (resp., increase) $d(i,j)$, if the rank of $d(i,j)$ is larger (resp., smaller) than that of $\delta(i,j)$. After moving the points according to the 'forces', the new 'forces' are computed again, and the whole adjusting process of the OTUs in $\mathcal{R}$ is iterated until the positions of OTUs converge sufficiently.

In this 'overdamped dynamics' the point configuration is driven by the potential energy

$$\Delta \equiv \sum (T_n - n)^2, \tag{1}$$

where T_n is the actual rank of the distance between the pair in $\mathcal{R}$ whose dissimilarity has the true rank n in the original data. $\Delta = 0$ is the ideal case. This Δ may be regarded as a counterpart of the stress in the conventional NMDS. As we will see in the next section, we can use quantities related to Δ to evaluate the confidence level of the resultant configuration statistically. Now, we can set up a null hypothesis to reject at a given confidence level. Furthermore, we can even discuss the plausibility of the substructures of the resultant configuration.

Thus an important feature of our NMDS algorithm is that the optimization process is directly connected to a process that improves the confidence level of the resultant configuration.

5 Measure of Confidence Level

Our algorithm is not free from the problem of local minima as all of the previously proposed algorithms. Generally speaking, NMDS cannot give the perfect solution with $C_{ij} = 0$ for all pairs (i,j). We usually have several local minimum solutions depending on initial trial configurations.

In the conventional NMDS [2], stress is used to judge the plausibility of the obtained configuration. Since we do not have such an extra device as $\hat{d}$, we cannot use stress to check the quality of the result. Our measure of discrepancy Δ defined by (1) is statistically not trivial (because d_{ij} are not independent, even if the positions of the OTUs are), and we do not know its statistical property. However, the following closely related quantity can be defined for each OTU j

$$\Delta(j) \equiv \sum [T_n(j) - n]^2, \tag{2}$$

where $T_n(j)$ and n are rank order only within $N-1$ (i,j) pairs for the given j. Thus, $\Delta(j)$ can be regarded as a statistical variable for the relative position of the jth OTU with the remaining OTUs. We can estimate the probability $P(d)$ of $\Delta(j) < d$ with the null hypothesis that the rank ordering of d_{ij} $(i \in \{1,2,\cdots,N\} \setminus \{j\})$ is totally random with respect to the rank ordering of δ_{ij} $(i \in \{1,2,\cdots,N\} \setminus \{j\})$. If N is sufficiently large, then $\Delta(j)$ obeys a normal distribution $\mathrm{N}((M^3-M)/6, M^2(M+1)^2(M-1)/36)$ $(M \equiv N-1)$. For smaller N there is a table for $P(d)$ [6]. Thus we can always test the null hypothesis with a given confidence level for jth OTU.

Contrary to Takane's approach[7], our method is nonparametric (i.e., model free). The nonparametric statistical properties of the stress has never been studied for NMDS to our knowledge.

As can easily be noted, instead of Δ defined in (1), we could use $\sum_j \Delta(j)$ (or an appropriately weighted sum) or the corresponding confidence levels as the potential function for the dynamics. This algorithm will be studied separately elsewhere.

One should notice that our criterion can be used for conventional NMDS with $\hat{d}$. The difference between ours and the currently popular ones is that our NMDS uses an optimization procedure directly related to the improvement of the confidence level of the resultant configuration.

6 Molecular Biological Example

To illustrate the use of our algorithm we deal with DNA sequence data of cichlid fish in both Lake Tanganyika and Lake Malawi [8]. For simplicity δ_{ij} is defined as the number of base (ATGC) mismatches, i.e.,

$$\delta_{ij} = L - \sum_k \delta[s_{ik}, s_{jk}],$$

where s_{ik} is the base (ATG or C) at kth position of the DNA sequence of OTU i and $\delta[s_{ik}, s_{jk}]$ takes 1 only when $s_{ik} = s_{jk}$ and L is the total number of

bases. One might criticize that this dissimilarity is not a reasonable choice, because usually the number of uncommon bases is not proportional to the time since the speciation occurred. However, since what we need is just a rank order, any definition which does not alter the rank order gives the same result. Thus we believe that our results obtained by NMDS is robust and is not affected by a particular definition of distance between the sequences. In this example, $N = 31$. First, we try to imbed OTUs into $2D$ Euclidean space. Five trial runs had very different values of $\Delta(j)$

All solutions have a few worse-than-1%-level OTUs. Thus, as a whole, $D = 2$ cannot be regarded as a good imbedding dimension. However, if we exclude 24, 27, and 28th OTUs from solution 5, the solution can be regarded as a good $2D$ configuration with the confidence level better than 0.5%. Or, if we construct more solutions, there may be a better solution. This is just an illustration, so we do not go further in this paper.

For a $3D$ Euclidean space as the imbedding space 27th OTU was with 1% confidence level and for three out of five runs 30th OTU was with 0.5% confidence level. Thus again, we cannot get any solution with confidence level better than 0.5%, but if we exclude 27th and/or 30th OTUs, we can get such a solution for the remaining OTUs.

Even for $D = 4$, we cannot get 0.5% confidence level solution for all OTUs. In two out of five runs, 24th OTU is with 1% confidence level, and in the remaining three runs 27th OTU is with 0.5% confidence level.

Our result is consistent with the phylogenetic analysis due to Hasegawa and Kishino [9], where some phylogenetic clades are reported. For example, $\{1, 2, \cdots, 9\}$, $\{10, 11, \cdots, 14\}$, and $\{15, 16, \cdots, 21\}$ are major clades. All of them can be seen as clusters in our results

However, we do not claim that our results reproduce phylogenetic analysis fully. For example, the main purpose of [9] is to show that the clade $\{1, 2, \cdots, 5\}$ is monophyletic, which can never be seen in our results. In the phylogenetic analysis, even if genetic distance is short, OTUs cannot be regarded as neighbors when there are side branches between them. On the other hand, in our analysis, short genetic distance means neighboring OTUs. Thus, what we see can differ from phylogenetic relations. The biological meaning of this possible discrepancy must be explored.

7 Summary and Concluding Remarks

We have proposed a novel algorithm for non-metric multidimensional scaling (NMDS) that is presumably the most faithful to the original idea of NMDS. This is why we call the proposed method the pure NMDS. The algorithm is closely connected to the nonparametric statistical confidence level of the resultant spatial configuration of the OTUs. Thus, we can use statistical criterion to evaluate the plausibility of the OTU configuration and its subconfigurations in the imbedded result. The numerical efficiency of our algorithm allows us to handle a large number (even 10,000 with the aid of an inexpensive desktop computer within a day) of OTU. This feature is worthy of stressing from the practical (esp., bioinformatics) point of view.

Acknowledgements

YO was supported, in part, by the National Science Foundation Grant DMR-99-70690.

References

[1] I. Borg and P. Groenen, (1997) *Modern Multidimensional Scaling* Springer, New York.

[2] R. N. Shepard and J. B. Kruskal (1964). Nonmetric Methods for Scaling and for Factor Analysis, *Am. Psychol.*, **19**, 557-558.

[3] T. F. Cox and M. A. A. Cox, (1994) *Multidimensional Scaling*, Chapman & Hall, London.

[4] P. E. Green, F. J. Carmone, JR. and S. M. Smith, (1970) *Multidimensional Scaling : Concepts and Applications*, Allyn and Bacon, Massachusetts.

[5] M. W. Trosset (1998) A New Formulation of the Nonmetric Strain Problem in Multidimensional Scaling, *J. Classification*, **15**, 15-35.

[6] E. L. Lehmann (1975) *Nonparametrics* Holden-Day, Inc., San Francisco.

[7] Y. Takane (1978) *Multidimensional Scaling Method* (in Japanese) Tokyo University Press, Tokyo.

[8] T. D. Kocher, J. A. Conroy, K. R. McKaye, et al, (1995). Evolution of the ND2 gene in East African cichlids, *Mol. Phyl. Evol.*, **4**, 420-432.

[9] M. Hasegawa and Y. Kishino, (1996) *Molecular Phylogeny* (in Japanese), Iwanami, Tokyo.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Optimal Design for the Evolution of Composite Mappings

Hideaki Suzuki
ATR International, Information Sciences Division
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288 Japan
hsuzuki@isd.atr.co.jp

As an example of the optimization of an evolutionary system design, a composite mapping problem is studied. In this problem, the system's basic design is determined by a set of elementary characters and their functions (mappings). Based on a certain definition for the functionality of a phenotypic mapping, a modified measure of system evolvability is proposed, and by maximizing this measure, the elementary function (mapping) set is optimized. A numerical experiment is also conducted in which the evolution is simulated using a set of genotypic entities, and the functional evolvability after optimization is compared to that before optimization.

Keywords: *evolvability, protein, amino-acid design, genotype-phenotype mapping, composite mapping*

1 Introduction

In the design of an artificial life (alife) system, the system's 'evolvability', defined as the possibility (potential ability) of evolving a variety of genotypes or phenotypes, is one of the most important properties a designer has to pay attention to [1, 2, 3, 4, 5, 6]. The evolvability determines the system's final performance; hence to design a 'good' alife system, we have to answer the following three primary questions about evolvability.

- What is an objective measure for evolvability?
- What are the necessary conditions for an alife system to be highly evolvable?
- How can we optimize an alife system design in terms of evolvability?

Recently, focusing on the evolvability of natural/artificial proteins, the author initiated a study addressing the first two questions [7, 8, 9]: a new evolvability measure ρ, defined as the density of functional genotypes in the protein genotype space, was proposed, and a necessary condition for high evolvability $\rho > \rho_c$ was established, where ρ_c is a critical density above which the functional protein genotypes are strongly connected in the genotype space with unit mutational modifications.

Based on this previous study, this paper addresses the third question by presenting a fictional protein model in which the protein's function is a mapping on a finite integer set. The protein's genotype is a sequence of elementary characters (amino acids) whose functions are defined as mappings on the integer set, and the protein's phenotype is a mapping created by the composition of the elementary mappings specified by the genotypic characters. For a definition of the functionality of the protein phenotype, a certain number of objective mappings is prepared by random choice, and a mapping is judged to be functional if it is 'close' to one of these objective mappings.

The basic system design of this model is determined by the elementary character set (amino acid set) and their functions (mappings). The distribution of functional genotypes in the genotype space depends upon this elementary set, so that to make evolutionary search in the genotype space to find a variety of functions, we have to optimize the elementary set so that a variety of functional genotypes might be present and interconnected in the genotype space. For this purpose, the author introduces a modified measure of evolvability, represented by the sum of the density of functional genotypes in the genotype space and a term proportional to the minimum size of a set of genotypes with the same objective function. Experiments are conducted in which the elementary mapping set is optimized using this measure, and champion data for the optimization are presented. An evolutionary simulation using a population of protein genotypes is also conducted, and the improvement of evolvability from the initial design is confirmed.

The organization of the paper is as follows. In Sect. 2, the fictional protein model for the composite mapping problem is introduced and the experimental methods are described. The results are presented in Sect.

3, and finally Sect. 4 summarizes the conclusion and future research plans.

2 The Model

Elementary mapping set: Throughout this paper, a mapping defines a one-to-one correspondence from an integer set $\{1, 2, \cdots, L\}$ to the same integer set $\{1, 2, \cdots, L\}$. A mapping is represented by an $L \times L$ mapping matrix whose (l_1, l_2)th element is 1 if integer l_1 is mapped to integer l_2 and 0 otherwise. For example, mapping $\{1 \to 2, 2 \to 4, 3 \to 3, 4 \to 1\}$ (for $L = 4$) is represented by the following 4×4 mapping matrix

$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}.$$

Note that every column and row of a mapping matrix have only one 1. As a basic design of the system, we prepare K elementary mappings and express them by characters such as $\mathrm{A}, \mathrm{B}, \mathrm{C}, \mathrm{D}, \cdots$. So, the design is represented by a set like

$$\mathrm{A} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, \mathrm{B} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \cdots.$$

Genotype and phenotype: A genotype of this protein model is a sequence of I elementary characters like "BAAABA" (for $I = 6$). The phenotype of a particular genotype is a mapping created by the composition of elementary mappings and is represented by the product of the elementary matrices. For example, the phenotype of "BAA" ($I = 3$) is represented by

$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

Objective mapping set and functionality: For a definition of the functionality of a phenotypic mapping, an objective mapping set is prepared which comprises J constant mappings chosen randomly. Any two mappings are judged to *fit* each other if and only if the number of integers mapped differently by the two is smaller than or equal to R, and a mapping is judged to be *functional* if and only if it fits one of the objective mappings.

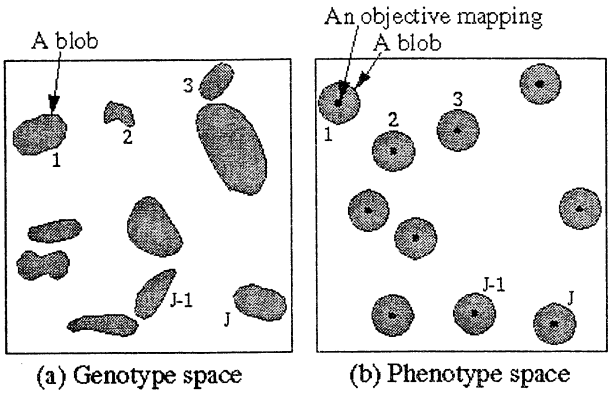


Figure 1: Symbolized figures of (a) the whole genotype space (the sequence space whose size is K^I) and (b) the whole phenotype space (the mapping space whose size is $L!$). In both figures, grey regions represent functional genotypes/phenotypes, and white regions represent nonfunctional genotypes/phenotypes. The phenotypic blobs, namely the sets of adjacent functional phenotypes have a uniform size (whose radius is R), whereas the genotypic blobs have various sizes.

Modified evolvability measure: Figure 1 symbolically shows the distribution of functional genotypes/phenotypes in the protein genotype/phenotype space. The distribution of the phenotypic blobs is determined only by the objective mapping set, whereas the distribution of the genotypic blobs is influenced by both the objective mapping set and the elementary mapping set. The evolvability measure ρ (the density of functional genotypes in the genotype space) [7, 8, 9] is formulated as

$$\rho = \sum_{j=1}^{J} \rho_j, \tag{1}$$

where

$$\rho_j = [\text{size of the } j\text{th genotype blob}]/K^I. \tag{2}$$

On the other hand, no matter how large ρ is, if the genotype blob sizes are unevenly distributed, we cannot expect an evolutionary search on the genotype space to

find a number of different functions (different objective mappings). To remedy this problem, the author introduces a modified measure of evolvability $\tilde{\rho}$ defined as

$$\tilde{\rho} = \sum_{j=1}^{J} \rho_j + J \cdot \min_{1 \le j \le J} \rho_j. \quad (3)$$

The second term of $\tilde{\rho}$ is proportional to the minimum size of the genotype blobs and prevents the optimization from being trapped by a bad elementary mapping set on which only a few genotype blobs are huge and the others are tiny.

Optimization procedure: Starting from an initial random elementary mapping set, the elementary set is revised so that $\tilde{\rho}$ might be maximized. For each revision, a randomly chosen transposition is operated on one of the elementary mappings, and if this modification enhances $\tilde{\rho}$, the elementary set is revised. In this paper, we limit the genotype space size K^I to a rather small value and evaluate ρ_js exactly by testing all genotypes in the genotype space.

Assessment by an evolutionary experiment: In order to compare the evolvability for the elementary mapping set after optimization to the evolvability before optimization, evolutionary experiments are conducted on both the initial (random) elementary mapping set and the optimized elementary set. Each run starts from an initial population made up of only one individual whose genotype is functional and simulates generation cycles by operating mutation and selection alternately on the population. (Mutation substitutes a randomly chosen character of a randomly chosen individual with a random character, and selection creates a new population using a roulette wheel in which functional individuals have fitness one and nonfunctional individuals have fitness zero.) Every generation, the number of objective mappings that have emerged during a run is evaluated, and when this number does not increase for a consecutive hundred generations, the number is judged to be saturated and a performance parameter d, the ratio of this saturated number to J, is calculated. The experiment is conducted twenty times using different random number sequences for each mapping set, and the average value of d is calculated as a result.

3 Results

Table 1 shows data for the best elementary mapping set obtained from about one hundred optimization runs. With this optimization, both the minimum genotype blob size ($K^I \cdot \min \rho_j$) and the density of functional genotypes (ρ) increase with an increase in the modified evolvability measure ($\tilde{\rho}$). ρ's value after optimization is much higher than its initial value of 0.012, which is equal to the theoretical value of ρ_{p} defined as the density of functional phenotypes in the phenotype space. See Appendix A for a theoretical estimation of ρ_{p}.

	Before optim. (Random elementary set)	After optim. (Optimized elementary set)
$K^I \cdot \min \rho_j$	5	10
$\rho = \sum \rho_j$	0.0120	0.0184
$\tilde{\rho} = \sum \rho_j + J \min \rho_j$	0.0188	0.0321
d	0.290	0.505

Table 1: Champion data for the optimization of the elementary mapping set. The upper three rows are the values calculated from the exact values of ρ_js, and the lowest data represents the result from evolutionary simulations. The parameter values are $L = 10$, $R = 4$, $J = 20$, $K = 11$, and $I = 4$.

The lowest row of Table 1 shows the result from evolutionary experiments on the elementary mapping sets before and after the optimization. We can conclude from this result that the protein evolvability was definitely improved by the optimization of the elementary mapping set.

4 Discussion

Using a model of fictional proteins whose functions are mappings on an integer set, an elementary amino-acid set was optimized in terms of the maximization of a modified measure of evolvability. The modified measure reflects the density of functional genotypes in the protein genotype space and the minimum size of genotype blobs. The latter factor assures the diversity of functions obtained by an evolutionary search on the genotype space. A direct simulation of the evolution of proteins was also conducted, and it was confirmed that the protein evolvability was enhanced by the optimization.

Future plans to extend the study are as follows.

Optimization for a larger problem: Although in the present experiment, ρ_js (Eq. (2)) were exactly evaluated with the whole genotype space (size is $K^I = 11^4 = 14641$), if we wish to estimate ρ_js with a larger I, we should evaluate ρ_js numerically using an approxima-

tion method like the Monte Carlo method. Establishing an optimization procedure under such a noisy measure will be a problem in the future. Also, for a general problem, we do not know J (the number of objective functions in the phenotype space) in advance. The maximization measure $\tilde{\rho}$ (Eq. (3)) must be modified when considering such a general problem.

Optimizing a language design for an alife system: As was shown in Fig. 1 and the subsequent analyses, system evolvability is principally determined by the genotype-phenotype mapping of proteins, or in other words, a language that specifies the correspondence between a sequence of symbols (characters) and its function. Here, a symbol (a character) denotes an object or an object manipulation in the world wherein the proteins act, and using such an object set, we have to design an appropriate symbol set to make a protein highly evolvable. For a machine language alife system [10, 11, 12], a symbol denotes an arithmatic operation of a register or a memory data word, and a symbol set must be optimized so that the final protein functions might contribute to self-replication (the final purpose of a creature) as much as possible. Optimizing such a protein language is also a future problem.

Acknowledgements

Dr. Eggenberger and Dr. Maeshiro of ATR-ISD labs provided helpful discussions for this study. Dr. K. Shimohara of ATR labs also actively encouraged this research.

A Theoretical estimation of ρ_{p}

Let $M(L,r)$ be the number of mappings on the integer set $\{1,2,\cdots,L\}$ that map r integers differently from a particular mapping. Because the phenotype blob size is given by $\sum_{r=0}^{R} M(L,r)$, if we neglect the overlapping between phenotype blobs, we can formulate ρ_{p} (the density of functional mappings in the phenotype space) as

$$\rho_{\mathrm{p}} = \frac{J}{L!}\sum_{r=0}^{R} M(L,r). \qquad \text{(A1)}$$

$M(L,r)$ is calculated using a recursion formula. We start from the initial condition $M(0,0) = 1$. Because the number of choices of r integers mapped differently from a particular mapping out of l integers is $\binom{l}{r}$, $M(l,r)$ satisfies

$$M(l,r) = \binom{l}{r} \cdot M(r,r). \qquad \text{(A2)}$$

Also, since the number of all mappings on l integers is $l!$, $M(l,r)$ satisfies

$$\sum_{r=0}^{l} M(l,r) = l!. \qquad \text{(A3)}$$

Using Eqs. (A2) and (A3), we can calculate $M(l,r)$ recursively by increasing l one by one. For $L = 10$, $R = 4$, and $J = 20$, the above procedure gives $\rho_{\mathrm{p}} = 0.01199$.

References

[1] Altenberg, L.: Evolvability checkpoints against evolutionary pathologies. In: Maley, C.C., Boudreau, E. (eds.): Artificial Life 7 Workshop Proceedings (2000) 3–7

[2] Bedau, M.A., Packard, N.H.: Measurement of evolutionary activity, teleology, and life. In: Langton, C.G. et al. (eds.): Artificial Life II: Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems (Santa Fe Institute Studies in the Sciences of Complexity, Vol. 10). Addison-Wesley (1992) 431–461

[3] Bedau, M.A., Snyder, E., Packard, N.H.: A Classification of Long-Term Evolutionary Dynamics. In: Adami, C., Belew, R., Kitano, H., and Taylor, C. (eds.): Artificial Life VI: Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems. MIT Press, Cambridge (1998) 228-237

[4] Taylor, T.: On Self-reproduction and Evolvability. In: Floreano, D. et al. (eds.): Advances in Artificial Life (5th European Conference on Artificial Life Proceedings) Springer-Verlag, Berlin (1999) 94-103

[5] Ray, T.S., Xu, C.: Measures of evolvability in tierra. In: Sugisaka, M., Tanaka, H. (eds.): Proceedings of The Fifth International Symposium on Artificial Life and Robotics (AROB 5th '00) Vol. 1 (2000) I-12-I-15

[6] Nehaniv, C.L., Rhodes, J.L.: The Evolution and Understanding of Biological Complexity from an Algebraic Perspective. Artificial Life **6** (2000) 45–67

[7] Suzuki, H.: Minimum Density of Functional Proteins to Make a System Evolvable. In: Sugisaka, M., Tanaka, H. (eds.): Proceedings of The Fifth International Symposium on Artificial Life and Robotics (AROB 5th '00) Vol. 1 (2000) 30-33

[8] Suzuki, H.: Evolvability Analysis Using Random Graph Theory. Proceedings of AFSS 2000 (The Fourth Asian Fuzzy Systems Symposium) Vol. 1 (2000) 549-554

[9] Suzuki, H.: Evolvability Analysis: Distribution of Hyperblobs in a Variable-Length Protein Genotype Space. In: Bedau, M.A. et al. (eds.): Artificial Life VII: Proceedings of the Seventh International Conference on Artificial Life. MIT Press, Cambridge (2000) 206–215

[10] Ray, T.S.: An approach to the synthesis of life. In: Langton, C.G. et al. (eds.): Artificial Life II: Proceedings of an Interdisciplinary Workshop on the Synthesis and Simulation of Living Systems (Santa Fe Institute Studies in the Sciences of Complexity, Vol. 10). Addison-Wesley (1992) 371–408

[11] Suzuki, H.: An Approach to Biological Computation: Unicellular Core-Memory Creatures Evolved Using Genetic Algorithms. Artificial Life **5** N.4 (2000) 367–386

[12] Suzuki, H.: Evolution of Self-reproducing Programs in a Core Propelled by Parallel Protein Execution. Artificial Life **6** N.2 (2000) 103–108

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Computation and Life in a Reversible Cellular Space

Kenichi MORITA, and Katsunobu IMAI
Faculty of Engineering, Hiroshima University
Higashi-Hiroshima, 739-8527, Japan
{morita, imai}@ke.sys.hiroshima-u.ac.jp

Abstract

A reversible cellular automaton (RCA) is a "backward deterministic" CA in which every configuration of a cellular space has at most one predecessor. It is an abstract model of a physically reversible space. In spite of the strong constraint of reversiblity, they have very rich ability of computing, and various interesting life-like phenomena can emerge in such a cellular space. We first discuss computation-universality of RCAs, and show several simple 2-D RCA models having this property. Next we discuss the problem of self-reproduction in RCAs. We describe our previous model of 2-D RCA in which a variety of objects can reproduce themselves by "shape-encoding" mechanism. We also show our recent RCA model of 3-D self-reproduction.

Keywords: self-reproduction, cellular automata, reversible computing, computation-universality

1 Introduction

A reversible cellular automaton (RCA) is an artificial and abstract model of space and time reflecting physical reversibility. Since our physical world is quantum mechanical (in a microscopic level) and its evolution is reversible, it is reasonable and important to investigate RCAs and their information processing abilities as well as related spatiotemporal phenomena. On the other hand, it is well known that von Neumann [1] first designed a self-reproducing CA, i.e., he showed it is possible to construct a universal computer (Turing machine) that can reproduce itself in a 29-state 2-D (irreversible) cellular space. In this paper, we investigate and discuss the problems of how the functions of computing and self-reproduction can be realized in a reversible environment, especially in a reversible cellular space. As we shall see, in spite of the strong constraint of reversibility, RCAs have rich abilities of computing and self-reproduction.

As for the computing ability of RCAs, Toffoli [2] first showed that a 2-D RCA is computation-universal. Later, Morita and Harao [3] strengthened his result by showing that a 1-D RCA is universal. For the case of 2-D RCA, Margolus [4] proposed an interesting 2-state model with so-called Margolus neighborhood which is a little different framework from the usual CA. After that, several simple universal models of 2-D RCAs using the framework of a partitioned CA (PCA) have been given [5, 6, 7, 8]. From these results we can see that universal computers can be constructed based on very simple and primitive reversible rules. Furthermore, in some cases, computation in RCAs can be carried out in a very different way from that of conventional computers. Thus it may lead new insights for future architectures of computing. Here, we discuss how universal computing mechanisms can be decomposed into such simple reversible rules.

Self-reproduction is also possible in an RCA. At present, we assume the so-called Langton's criterion [9], i.e., self-reproducing objects need not have computation-universality, but the construction of a daughter object should be actively directed by its mother by using its "gene" properly. We first show our previous model of 2-D self-reproducing RCA called SR_8 [10], where various shapes of objects called Worms and Loops can self-reproduce. Such objects have abilities of (1) encoding the shape into a "gene" represented by a command sequence (it is called a "shape-encoding" mechanism; see also [11]), (2) copying the gene, and (3) interpreting the gene to create a shape. We can design so that these operations are all performed reversibly. Furthermore, because of the shape-encoding mechanism, a Worm or a Loop of almost any shape can self-reproduce in a very simple fashion. We also describe our recent attempt to create a 3-D self-reproducing RCA (SR_9) [12]. Its basic mechanism is similar to that of the 2-D case. However, it is not a mere extension of the 2-D model, because varieties of possible shapes and arrangements of Worms and Loops are much greater than that of the 2-D case.

2 Computation in RCAs

2.1 A partitioned CA (PCA)

Here, we use a framework of a partitioned CA (PCA) [3], which is a subclass of usual CAs. It greatly helps to design an RCA, because, in a PCA, injectivity (reversibility) of a global map is equivalent to injectivity of its local map [3]. A 2-D PCA with von Neumann neighborhood is a one whose cell is divided into five parts as shown in Fig.1.

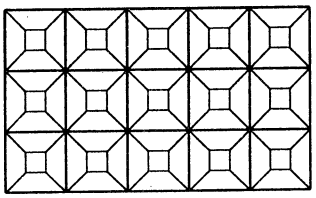


Figure 1: Cellular space of a 5-neighbor PCA.

The next state of each cell is determined by the present states of the center part of this cell, the lower part of the upper cell, the left part of the right cell, the upper part of the lower cell, and the right part of the left cell (not depending on the entire five cells). Hence, each transition rule is depicted as in Fig.2.

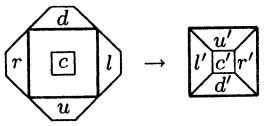


Figure 2: A rule of a 5-neighbor PCA.

Note that, in the case that the center part has only one state (thus it is actually the 4-neighbor case), the center part is omitted in a figure.

2.2 Simple universal 2-D RPCAs

If we want to construct a computation-universal *irreversible* 2-D CA, it is sufficient to construct a one such that conventional logic gates AND, OR, and NOT as well as routing of signals can be realized in its cellular space. However, since AND and OR are irreversible gates (in the sense that inputs are not uniquely determined from its output), it is not possible to design an RCA that directly realizes these gates. Instead, we take another approach, i.e., we construct RCAs that can embed a universal reversible logic element.

A Fredkin gate (F-gate) [13] is a reversible (i.e., its logical function is one-to-one) and bit-conserving (i.e., the number of 1's is conserved between inputs and outputs) logic gate shown in Fig.3. It is known as a universal reversible gate [13].

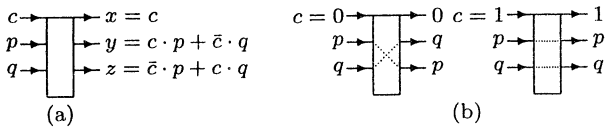


Figure 3: (a) A Fredkin gate, and (b) its function.

Two universal models of 4-neighbour 2^4-state RPCAs (called S_1 and S_2), in which F-gate can be embedded, have been given in [6]. Fig.4 shows the set of transition rules of S_1 (note that since S_1 is rotation-symmetric, rotated rules are omitted). Fig.5 shows a realization of a switch-gate (S-gate), a 2-input 3-output reversible gate. Since an F-gate can be constructed by switch-gates [13], S_1 realizes an F-gate.

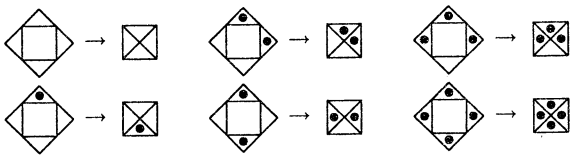


Figure 4: Transition rules of the universal RPCA S_1.

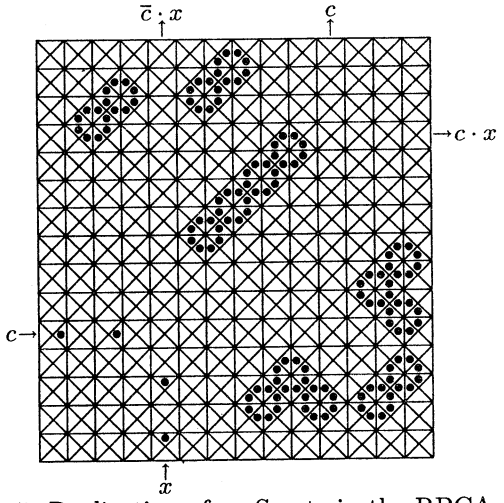


Figure 5: Realization of an S-gate in the RPCA S_1.

It is also possible to construct RPCAs having other types of grids. Imai and Morita [5] gave a universal 3-neighbor 2^3-state "triangular" RPCA T_1. T_1 has extremely simple transition rules as shown in Fig.6, and can realize an F-gate.

Besides an F-gate, a reversible logic element called a "rotary element" (Fig.7) is also known to be universal [7, 14]. It has been shown that a 4-neigbor 4^4-state RPCA model P_4 [6] and a 3^4-state model P_3 [8] can embed a rotary element. In these models, any reversible counter machines can be embedded very concisely. Furthermore, its design method is very different from the conventional one.

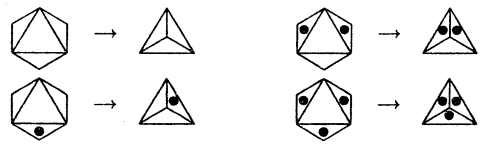


Figure 6: Transition rules of the triangular RPCA T_1.

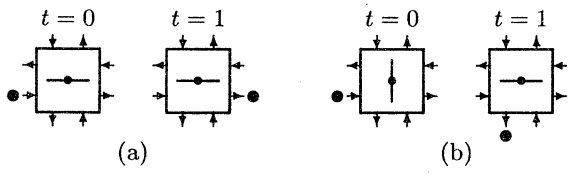


Figure 7: Operations of a rotary element: (a) the parallel case (the coming direction of a particle is parallel to the rotating bar), and (b) the orthogonal case.

3 Self-Reproduction in RPCAs

3.1 2-D self-reproduction

A 2-D 5-neighbor self-reproducing RPCA called SR_8 has been proposed in [10]. Each cell of SR_8 has 8^5 states (i.e., each of five parts of a cell has 8-state).

As in the 29-state CA of von Neumann, SR_8 also makes use of a genetic code (description of the object's shape). That is, the body of a daughter object is constructed by interpreting the description. Furthermore, if the machine can encode its shape into a description by checking its body dynamically, there is no need to keep the entire description. In fact, there have been a few models that performs self-reproduction in such a manner [10, 11, 15, 16]. The method employed in SR_8 and a 2-D 12-state (irreversible) CA model SR_{12} [11] is called a "shape-encoding" mechanism. This method has the following advantages:

1. Complexity of a self-reproducing configuration is very low. For example, a Worm or a Loop with only 4 cells can self-reproduce.
2. A Worm or a Loop of an arbitrary shape can self-reproduce. Thus, self-reproducing ability of an object is relatively robust against distortion.
3. Since shape-encoding and decoding are performed directly and symmetrically, their mechanisms can be easily understood.

Furthermore, in this cellular space, three main operations (i.e., encoding the shape of an object into a gene represented by a command sequence, copying the gene, and interpreting the gene to create an object) are all performed *reversibly*.

A *Worm* is a simple signal wire with two open ends: a head and a tail. At the tail cell the shape of the Worm is "encoded" into commands and the tail retracts one by one. The commands are sent to the head along the wire. At the head of the Worm, commands are decoded and executed to extend the head. Therefore, it crawls in the space keeping its shape cyclically. By putting a "branch" command, which makes a head branch, a Worm can self-reproduce as in Fig.8.

A *Loop* is a simple closed signal wire. It can also reproduce itself in a similar way as in a Worm by extending a "constructing arm" as shown in Fig.9.

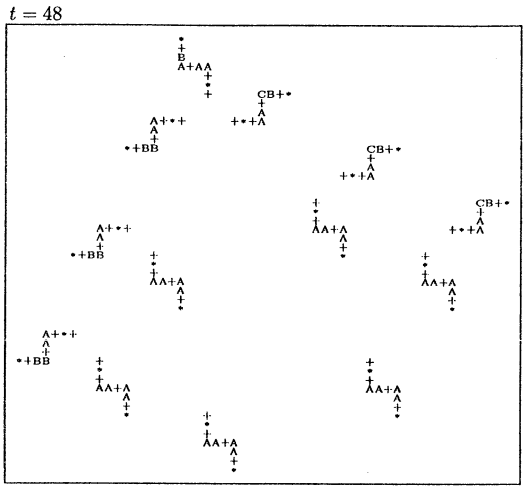


Figure 8: A self-reproducing Worm in SR_8.

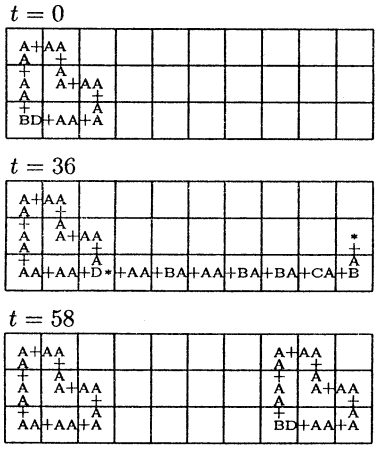


Figure 9: A self-reproducing Loop in SR_8.

3.2 3-D self-reproduction

By extending SR_8, we gave a 3-D self-reproducing RPCA SR_9 [12]. This model is a 7-neighbor RPCA, and has 9 states in each of seven parts of a cell (hence each cell has 9^7 states). As in SR_8, Worms and Loops of various shapes can reproduce themselves in SR_9.

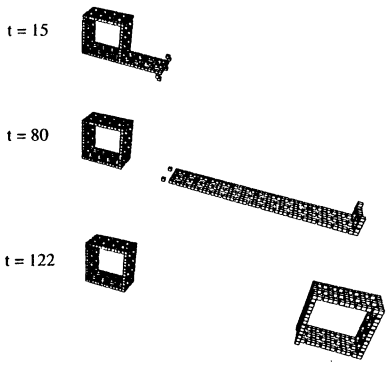


Figure 10: Self-reproduction of a 3-D Loop.

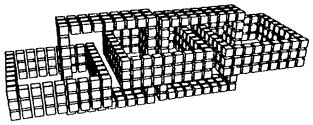


Figure 11: A chain formed by a self-reproducing Loop.

In the 3-D model, varieties of possible shapes and arrangements of Worms and Loops are much greater than that of 2-D. Fig.10 shows a simple self-reproduction process of a 3-D Loop. By controlling a position of a constructing arm by a command sequence, we can design a Loop such that it produces a semi-infinite chain as in Fig.11. Another example of a Loop that forms a pile of Loops is shown in Fig.12.

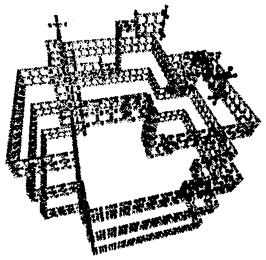


Figure 12: A more complex self-reproducing Loop.

4 Concluding Remarks

We have investigated how computing and self-reproduction tasks are performed in RCAs, and found that these tasks are decomposed into very primitive reversible rules. However, there remains a problem to construct an RCA having both computing and self-reproduction abilities in an elegant way.

Acknowledgements: This work was supported in part by Grant-in-Aid for Scientific Research (C) No. 12680353 from Ministry of Education, Science, Sports and Culture of Japan, and by the Okawa Foundation for Information and Telecommunication.

References

[1] von Neumann, J., *Theory of Self-reproducing Automata* (ed. A.W.Burks), The University of Illinois Press, Urbana (1966).

[2] Toffoli, T., Computation and construction universality of reversible cellular automata, *J. Comput. Syst. Sci.*, **15**, 213–231 (1977).

[3] Morita, K., and Harao, M., Computation universality of one-dimensional reversible (injective) cellular automata, *Trans. IEICE Japan*, **E-72**, 758–762 (1989).

[4] Margolus, N., Physics-like model of computation, *Physica*, **10D**, 81–95 (1984).

[5] Imai, K., and Morita, K., A computation-universal two-dimensional 8-state triangular reversible cellular automaton, *Theoret. Comput. Sci.*, **231**, 181–191 (2000).

[6] Morita, K., and Ueno, S., Computation-universal models of two-dimensional 16-state reversible cellular automata, *IEICE Trans. Inf. & Syst.*, **E75-D**, 141–147 (1992).

[7] Morita, K., Tojima, Y., and Imai, K., A simple computer embedded in a reversible and number-conserving two-dimensional cellular space, *Multiple-Valued Logic*, (to appear) (2000).
http://www.ke.sys.hiroshima-u.ac.jp/~morita/p4/

[8] Morita, K., and Ogiro, T., Embedding a counter machine in a simple reversible 2-D cellular space, *Proc. 6th IFIP Int. Workshop on Cellular Automata*, Osaka, 30–31 (2000).
http://www.ke.sys.hiroshima-u.ac.jp/~morita/p3/

[9] Langton, C.G., Self-reproduction in cellular automata, *Physica*, **10D**, 135–144 (1984).

[10] Morita, K., and Imai, K., Self-reproduction in a reversible cellular space, *Theoret. Comput. Sci.*, **168**, 337–366 (1996).
http://www.ke.sys.hiroshima-u.ac.jp/projects/rca/sr/

[11] Morita, K., and Imai, K., A simple self-reproducing cellular automaton with shape-encoding mechanism, *Artificial Life V* (eds. C.G. Langton and K. Shimohara), The MIT Press, 489–496 (1997).

[12] Hori, T., Imai, K., and Morita, K., Self-reproduction movies of 3D reversible cellular automata (1998).
http://kelp.ke.sys.hiroshima-u.ac.jp/projects/rca/sr3d/

[13] Fredkin, E., and Toffoli, T., Conservative logic, *Int. J. Theoret. Phys.*, **21**, 219–253 (1982).

[14] Morita, K., A new universal logic element for reversible computing, Technical Report of IEICE Japan, COMP99-94 (2000).

[15] Ibáñez, J., Anabitarte, D., Azpeitia, I., Barrera, O., Barrutieta, A., Blanco, H., and Echarte, F., Self-inspection based reproduction in cellular automata, in *Advances in Artificial Life* (eds. F. Moran et al.), LNAI-929, Springer-Verlag, 564–576 (1995).

[16] Laing, R., Automaton models of reproduction by self-inspection, *J. Theor. Biol.*, **66**, 437–456 (1977).

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A Design of Synchronization Algorithms for a Large Scale of Cellular Automata

Hiroshi Umeo, Jun Nishimura, Takashi Sogabe and Masashi Maeda

Osaka Electro-Communication University,
Graduate School of Engineering,
Faculty of Information Science and Technology,
Department of Engineering Informatics,
Neyagawa-shi, Hatsu-cho, 18-8, Osaka, 572-8530, Japan

Abstract In this paper, we study synchronization algorithms for a large scale of cellular automata. It is shown that there still exist several new synchronization algorithms with a smaller number of internal states for one and two-dimensional cellular arrays. We also give a survey of synchronization algorithms developed so far.

Keywords Cellular automaton, Firing squad synchronization problem, Optimum-time synchronization algorithm, 6-state 3n-step firing squad synchronization algorithm, 1-bit inter-cell communication cellular automata

1 Introduction

In recent years cellular automata have been establishing increasing interests in the study of modeling real phenomena occurring in biology, chemistry, ecology, economy, geology, mechanical engineering, medicine, physics, sociology, public traffic, etc. In this paper, we study firing squad synchronization algorithms which give a finite-state protocol for synchronizing a large scale of cellular automata. The synchronization in cellular automata is known as firing squad synchronization problem since the development of the problem, where it was originally proposed by J. Myhill to synchronize all parts of a self-reproducing cellular automata[1]. The firing squad synchronization problem has been studied extensively over 40 years and about 60 to 70 papers have been published[1-22]. In the next section, the firing squad synchronization problem is introduced and a variety of synchronization algorithms on one-dimensional arrays will be given in section 3. We present an optimized transition rule set for the first-in-the-world Waksman's firing scheme. It is shown that there exists a 6-state CA which can synchronize n cells in $3n - O(\log n) + O(1)$ steps. In section 4, a 9-state implementation for synchronizing two-dimensional arrays will be given.

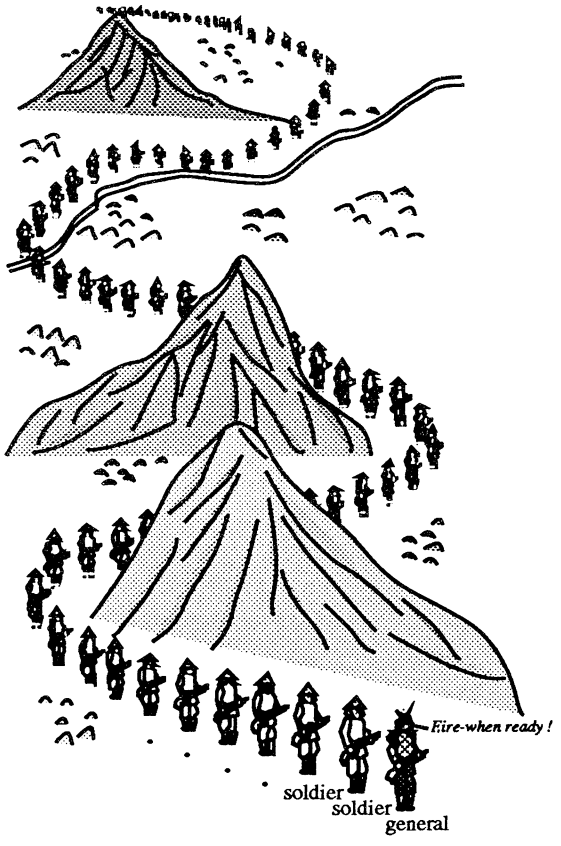


Fig. 1 Firing squad synchronization problem

2 Firing Squad Synchronization Problem

Please look at the Fig. 1. There are many soldiers arranged in line of battle in the desert. One

"*general*" is located at one end in the line and gives a command "*fire-when-ready*" to his nearest neighbor soldier at time $t = 0$. All soldiers are sleeping until the command will be delivered. In the desert, there are many high mountains and heavy sand storms are seen frequently, those make it difficult for soldiers and the general to communicate globally. Each soldier can only communicate with its nearest right and left neighbor soldiers, either does the general. The problem is to design the operations of soldiers so that, at some future time, all of them will simultaneously and, for the first time, enter a special "*firing*" state. The number of soldiers are not known to each soldier nor the general. The amount of internal memory of each soldier/general is finite state. Thus they cannot count the total number of the soldiers.

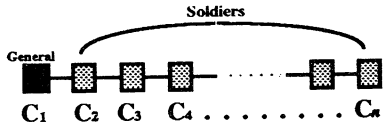


Fig. 2 One-dimensional cellular array

The firing squad synchronization problem is formalized in terms of the model of cellular automata. Fig. 2 shows a finite one-dimensional(1-D) cellular array consisting of n cells. Each cell is an identical (except the end cells) finite state automaton. The array operates in lock-step mode in such a way that the next state of each cell (except both end cells) is determined by both its own present state and the present states of its right and left neighbors. All cells ("*soldiers*"), except the left end, are initially in the quiescent state at time $t = 0$ with the property that the next state of a quiescent cell with quiescent neighbors is the quiescent state again. At time $t = 0$ the left end cell C_1 ("*general*") is in "*fire-when-ready*" state that is an initiation signal to the array. The firing squad synchronization problem is to give the description (state set and next-state function) of the cell such that all cells enter the "*fire*" state at exactly the same time and for the first time. The set of states must be independent of n. The signal can propagate down the line no faster than one cell per one step. Therefore $(n-1)$ steps are required to wake up the farthest soldier and additional $(n-1)$ steps are required for the general to receive the return signal from the farthest soldier. Thus, totally $(2n - 2)$ steps are necessary for firing n cells. The tricky part of the problem is that the same kind of soldier with a fixed number of states is required to synchronize, regardless of the length n of the array.

3 Synchronization Algorithms on 1-D Arrays

3.1 Optimum-Time Algorithm

In 1966, A. Waksman[4] proposed a 16-state firing squad synchronization algorithm, which is known, together with an unpublished Goto's[3] algorithm, as the first-in-the-world optimum-time firing algorithm. Balzer[5] and Gerken[8] improved the Waksman's scheme and presented 8- and 7-state firing squad synchronization algorithms, respectively. In 1987, Mazoyer[6] gave a 6-state optimum-time firing algorithm which is known as the best algorithm with the smallest number of states. We state and compare those algorithms in the following theorem and Table 1.

[Theorem 1] [3]-[8]There exists a one-dimensional CA which can synchronize n cells in exactly $2n-2$ steps.

Waksman described his algorithm in terms of a finite state transition table, however, it has been reported in the talks of cellular automata researchers that some fatal errors were included in the Waksman's transition table. Umeo, Sogabe and Nomura[7] corrected all errors included in his original transition table and gave a complete list of transition rules which yield successful firings for any length of array. In our correction, ninety-three percent reduction has been made in the number of Waksman's original transition rules. It has been shown that two-hundred and two rules are necessary and sufficient ones for the Waksman's optimum-time firing squad synchronization.

Algorithms	Internal States	Transition Rules
Waksman[4]	16	3208 (some errors included)
Goto[3]	many thousands	---- no implementation
Balzer[5]	8	182
Mazoyer[6]	6	120
Gerken[8]	7	118
Umeo, Sogabe and Nomura[7]	16	202

Table 1 A comparison between optimum-time synchronization algorithms

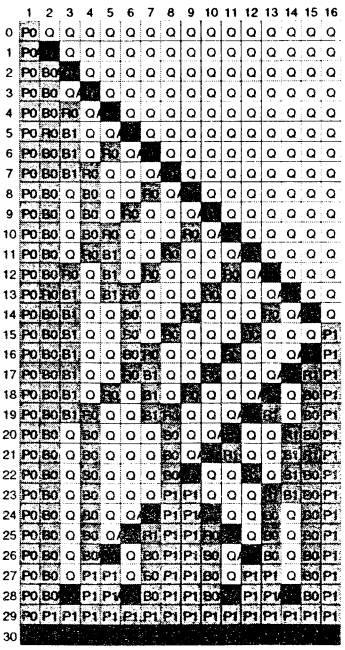


Fig. 3 A configuration for optimum-time Waksman's scheme based on our optimized transition rule set

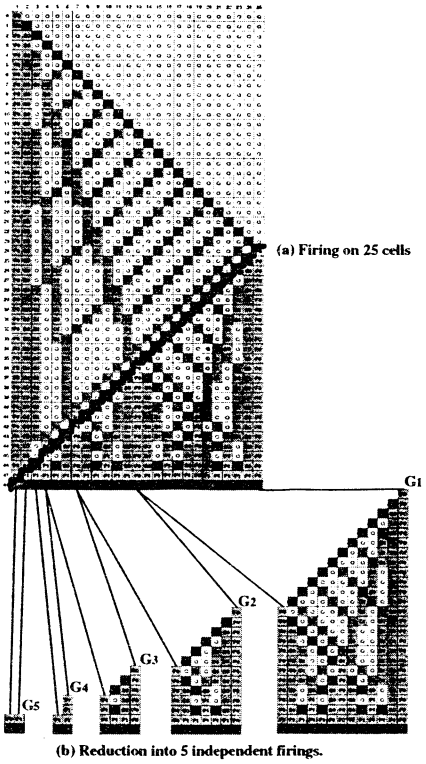


Fig. 4 Our new interpretation of Waksman's scheme. A firing for 25 cells can be reduced into 5 independent smaller subfirings.

Fig. 3 shows a firing configuration based on our optimized transition rule set. We have also given a formal proof of the validity of the transition rule set based on mathematical induction and presented a new interpretation of the Waksman's scheme. Fig. 4(a, b) shows our interpretation of firing processes with a help of an example of the firing configuration on 25 cells. The computations below the bold diagonal line consists of 5 independent firings for 5 sub-spaces which have 13, 7, 4, 2 and 2 cells. The last two firings are special ones on two cells which fire at 1 and 3 steps, controlled by the right-end *general*, respectively. Thus the firing on 25 cells can be reduced to 5 independent firings on smaller subspaces. What is more important is that each decomposed subfiring controlled by the right-end *general*, below the diagonal line, is a mirror image of the original firing controlled by the left end *general*.

3.2 Generalized Synchronization Algorithm

A generalized firing squad synchronization problem has been studied by several researchers[12-14], where the general is allowed to be located at any position in the line.

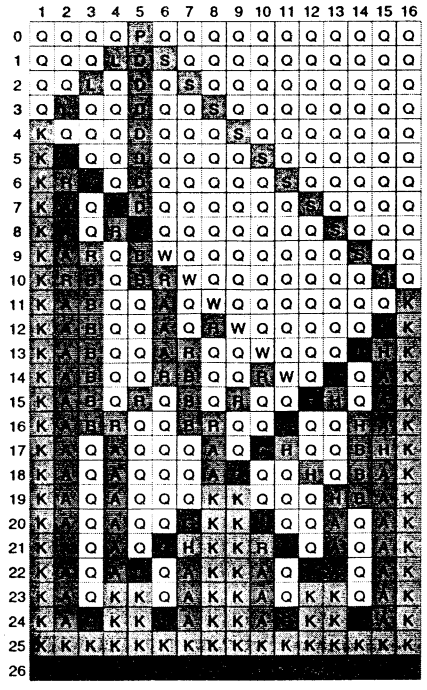


Fig. 5 A configuration for the generalized firing squad synchronization

Moore and Langdon [12] and Varshavsky, Marakhovsky and Peschansky[13] gave 17-state and 10-state optimum-time solutions, respectively. In Fig. 5 we show a configuration of generalized firing on 19 cells, where the general is on the 7th cell.

[Theorem 2] [12], [14] There exists a CA which can solve the generalized firing squad synchronization problem in optimum steps.

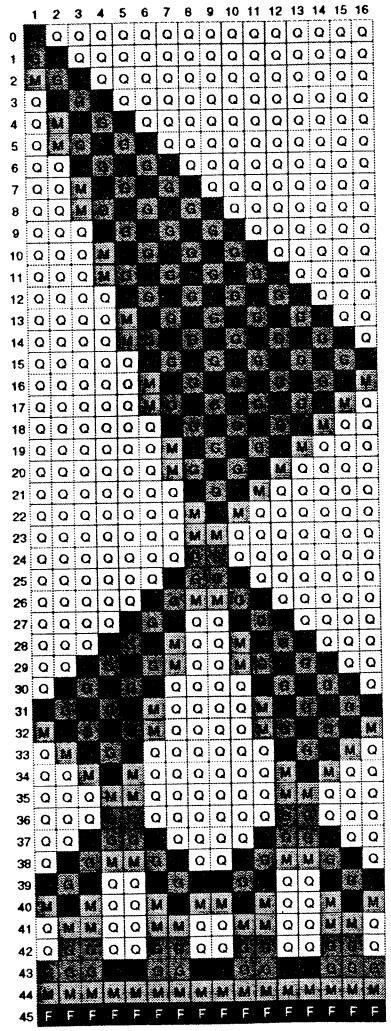


Fig. 6 A configuration for $3n$-step firing on n cells

3.3 Linear-Time Algorithm

In 1994, a French researcher J. B. Yunes[10] began to explore $3n$-step firing squad synchronization algorithms and developed two seven-state $3n$-step synchronization algorithms for the synchronization of any one-dimensional array of n cells. His algorithms were interesting, since he progressively decreased the synchronization algorithms for the synchronization of any one-dimensional array of n cells. Recently, Maeda and Umeo[11] proposed a new $3n$-step 6-state firing squad synchronization algorithm, which is an improvement of Yunes's results. The number "*six*" is the smallest one known at present.

[Theorem 3] [2], [10], [11] There exists a CA which can synchronize n cells in $3n - O(\log n) + O(1)$ steps.

3.4 1-Bit-Communication Synchronization Algorithm

In the long history of the study of cellular automata, the amounts of bit-information exchanged at one step between neighboring cells have been assumed to be $O(1)$-bit. Here we introduce a new class of cellular automata, $CA_{1\text{-bit}}$, whose inter-cell communication is restricted to 1-bit. First we consider the synchronization algorithm operating in linear-time on $CA_{1\text{-bit}}$. A $3n$-step firing squad synchronization algorithm given in [Theorem 4] can be designed based on a parallel divide-and-conquer strategy which utilizes 1/1- and 1/3-speed waves efficiently[16].

Algorithms	Internal States	Transition Rules	Time Complexity
Minsky and McCarthy(1967)	13	130	$3n + O(\log n)$
Fischer(1965)	15	?	$3n$ -4
Mazoyer(?)	14	?	$3n$ -1
Yunes(1993)	7	134	$3n + O(\log n)$
Yunes(1993)	7	134	$3n - O(\log n)$
Maeda and Umeo(2000)	6	79	$3n$ -3 + $O(\log n)$

Table 2 A comparison between $3n$-step synchronization algorithms

Our 1-bit algorithm has 48 states including right and left boundary states. Fig. 7 shows the snapshots for the 1-bit synchronization in the case $n = 19$.

[Theorem 4][16] There exists a $CA_{1\text{-bit}}$ which can synchronize n cells in $3n + O(\log n) - O(1)$ steps.

Now we develop our optimum-time firing squad synchronization algorithm on $CA_{1\text{-bit}}$. Mazoyer[15] has obtained a $(2n\text{-}2)$-step synchronization algorithm with only one bit of information exchanged. The validity of his algorithm was shown only by computer simulation. The object of our design is to give the algorithm which assures its validity for any n. The optimum 1-bit synchronization algorithm we develop is based on Waksman's optimum-time algorithm which has been

shown valid for any n in Umeo, Sogabe and Nomura[7]. Our design is more transparent and clearer in understanding of the correctness of the algorithm. The number of internal states in each cell is 78 and the total number of transition rules is 208. In Fig. 8, we show the snapshots for the 1-bit synchronization based on Waksman's scheme.

[Theorem 5] [15]-[17] There exists a $CA_{1\text{-bit}}$ which can synchronize n cells in optimum $2n$-2 steps.

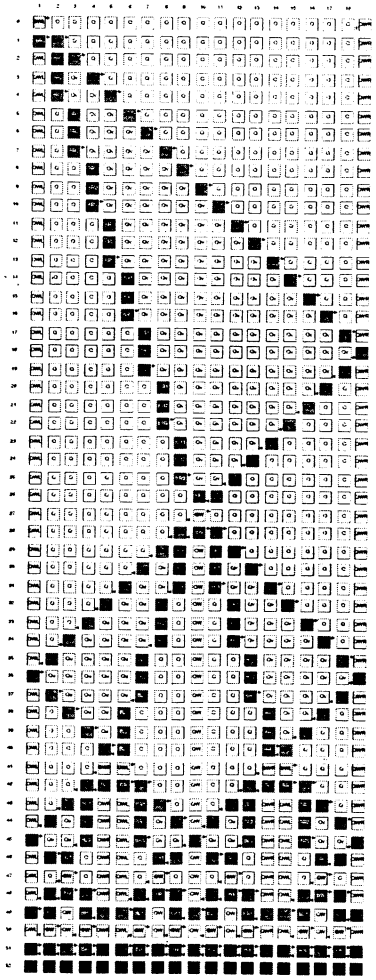


Fig. 7 Snapshots for 1-bit $3n$-step synchronization in the case n = 19

4 Synchronization Algorithms on 2-D Arrays

A two-dimensional cellular automaton consists of a 2-D finite array of identical finite state automata located at points in the plane with non-negative integer coordinates (i, j). Each automaton is referred to as a cell. A cell at point (i, j) is denoted by $C_{i,j}$ where i, j $\geqq$ 0. Each cell has a finite state memory. Each $C_{i,j}$ is connected with its north, west, south and east neighbor cells. Several 2-D synchronization algorithms and their implementations have been presented in Shinar[9], Grasselli[18], Beyer[19] and Szwerinski[20]. Fig. 9 shows our 9-state implementation on 2-D array.

[Theorem 6] [9], [18-21] There exists a 2-D CA which can synchronize n x n cells in optimum $2n$-2 steps.

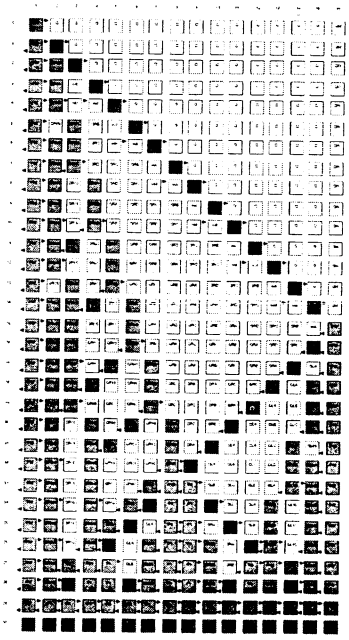


Fig. 8 A configuration of optimum-time 1-bit synchronization based on Waksman's scheme

5 Summary

We have given a survey of synchronization algorithms developed so far and proposed some new synchronization algorithms and implementations on one- and two-dimensional arrays.

Acknowledgments The authors would like to express their thanks to Mr. N. Kamikawa for his support on $CA_{1\text{-bit}}$ simulation software.

References

[1] E. F. Moore;"The firing squad synchronization problem", in "Sequential Machines" (E. F. Moore Ed.), Selected Papers, Addison-Wesley, Reading, MA, pp.213-214, (1964).
[2]M. Minsky;"Computation:Finite and infinite machines", Prentice Hall, pp.28-29, (1967).
[3] E. Goto;"A minimum time solution of the firing squad problem", dittoed course notes for Applied Mathematics

298, Harvard University, pp.52-59, with an illustration in color, (1962).
[4] A. Waksman;"An optimum solution to the firing squad synchronization problem", Information and Control, 9, pp.66-78, (1966).
[5] R. Balzer;"An 8-state minimal time solution to the firing squad synchronization problem", Information and Control, 10, pp.22-42, (1967).

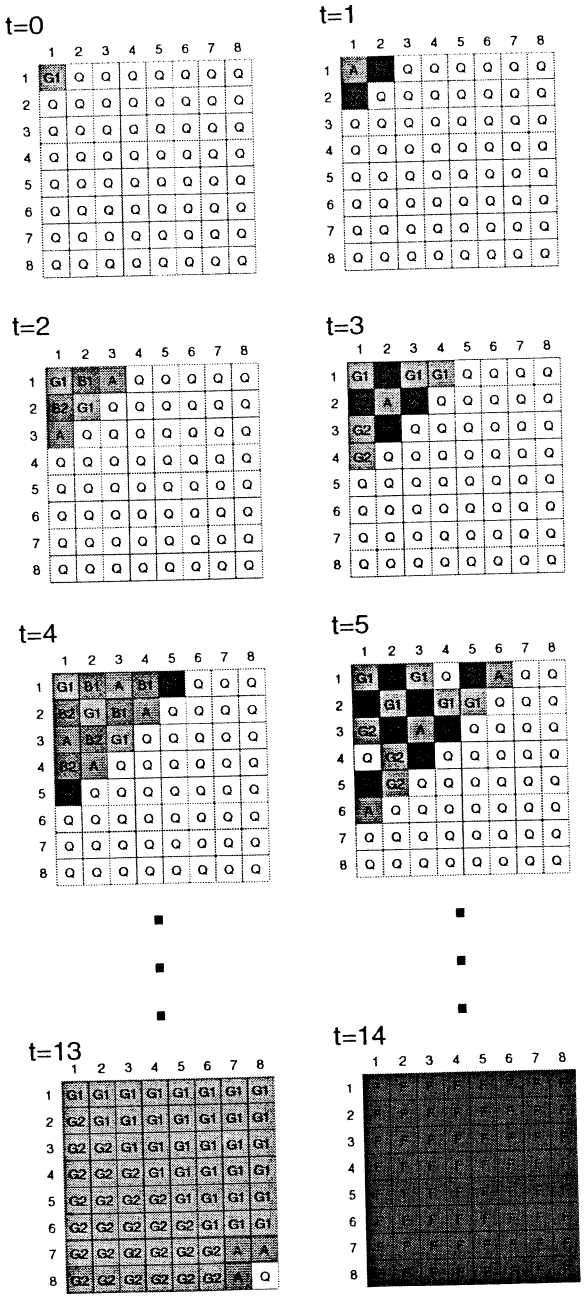


Fig. 9 A configuration of optimum-time synchronization on 8 x 8 2-D array

[6] J. Mazoyer;"A six-state minimal time solution to the firing squad synchronization problem", Theoretical Computer Science, 50, pp.183-238, (1987).
[7] H. Umeo, T. Sogabe and Y. Nomura;"Correction, Optimization and Verification of Transition Rule Set for Waksman's Firing Squad Synchronization Algorithm", Proc. of the Fourth Intern. Conference on Cellular Automata for Research and Industry", pp.152-160, (2000).
[8] Hans-D., Gerken;"Uber Synchronisations-Probleme bei Zellularautomaten", Diplomarbeit, Institut für Theoretische Informatik Technische Universitat Braunschweig, pp.50, (1987).
[9] I. Shinahr; "Two- and three-dimensional firing squad synchronization problems", Information and Control, 24, pp.163-180, (1974).
[10] J. B. Yunes;" Seven-state solutions to the firing squad synchronization problem", Theoretical Computer Science 127, pp.313-332, (1994).
[11] M. Maeda and H. Umeo;"A Design of 6-State 3n-Step Firing Squad Synchronization Algorithm and Its Implementation", Proc. of Domestic Conference on Artificial Intelligence held in Tokyo, (2000).
[12] F. R. Moore and G. G. Langdon;"A Generalized Firing Squad Problem", Information and Control, Vol. 12, pp.212-220, (1968).
[13] V. I. Varshavsky, V. B. Marakhovsky and V.A. Peschansky; "Synchronization of intercting automata ", Mathematical System Theory, Vol. 4, No.3, pp.212-230, (1970).
[14] V. I. Varshavsky;" Synchronization of a collection of automata with random pairwise interaction", Autom. and Remote Control, 29, pp.224-228, (1969).
[15] J. Mazoyer;"On optimal solutions to the firing squad synchronization problem", Theoret. Comput. Sci., 168, pp.367-404, (1996).
[16] H. Umeo, j. Nishimura and T. Sogabe;"1-Bit Inter-cell Communication Cellular Algorithms", (invited lecture), Proc. of the Tenth Intern. Colloquium on Differential Equations, held in Plovdiv in 1999, International Journal of Differential Equations and Applications, pp.433-446, Vol. 1A, No.4, (2000).
[17] J. Nishimura, T. Sogabe and H. Umeo;"A Design of Optimum-Time Firing Squad Synchronization Algorithm on 1-Bit Cellular Automaton", Tech. Rep. of IPSJ, (2000).
[18] A. Grasselli;" Synchronization of cellular arrays: The firing squad problem in two dimensions", Information and Control, 28, pp.113-124, (1975).
[19] W. T. Beyer;"Recognition of topological invariants by iterative arrays", Ph.D. Thesis, MIT, pp. 144, (1969).
[20] H. Szwerinski;"Time-optimum solution of the firing-squad-synchronization-problem for n-dimensional rectangles with the general at an arbitrary position", Theoretical Computer Science, 19, pp.305-320, (1982).
[21] M. Maeda and H. Umeo;"A 9-state implementation of two-dimensional firing squad synchronization algorithm", Proc. of Spring Joint Conference of IEICE, (to be presented), (2001).
[22] P. Rosenstiehl, J. R. Fiksel and A. Holliger; "Intelligent graphs:Networks of finite automata capable of solving graph problems", in "Graph Theory and Computing"(R.C.Reed Ed.), Academic Press, New York, pp.219-265, (1973).

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15–17, January, 2001,

Self-Timing in Biological Systems Simulated on Cellular Automata

Ferdinand Peper* peper@crl.go.jp
Teijiro Isokawa† tim@comp.eng.himeji-tech.ac.jp
Susumu Adachi* sadachi@crl.go.jp
Nobuyuki Matsui† matsui@comp.eng.himeji-tech.ac.jp

*Communications Research Laboratory, 588-2 Iwaoka, Iwaoka-cho, Nishi-ku, Kobe 651-2492, Japan
†Dep. Computer Engineering, Himeji Institute of Technology, 2167, Shosha, Himeji 671-2201, Japan

Abstract

Nature is asynchronous. There is no central clock in accordance to which the actions of molecules, proteins, cells, etc. are synchronized. Simulations of nature are usually synchronous. Cellular automata are an example of a synchronous model often used for simulations. This paper describes an (improved) cellular automaton model in which cells work asynchronously and whose transitions are triggered by their direct neighbors. Functions like wires, crossings, and AND-gates can be straightforwardly implemented on the model, but neural functions are also possible.

Keywords: asynchronous cellular automata, self-timing, chain reactions

1 Introduction

Cellular Automata (CA) have been used extensively to model systems in nature. For example, in theoretical biology they have found application in modeling self-reproduction [1, 2, 3], the emergence of patterns [4, 5], and other biological processes, such as biogradation [6]. They have also found use in modeling fluids [7] in physics, and in modeling systems in our human-made world, like the flow of pedestrians in train stations [8], etc. All these applications have in common that the synchronized operation of CA is not motivated by biological, physical, or other modeling considerations, but rather by some mathematical necessities and the wide availability of "synchronized logics", such as boolean functions, etc. [9].

Asynchronous cellular automata (ACA), on the other hand, are harder to deal with mathematically [10], because they behave in more unpredictable ways [11] than their synchronous counterparts, possibly reflecting the unpredictability of nature. They have the advantage, however, that less consideration is needed for attuning operations' timings of different parts of a system to each other. This would allow for simpler extension or modification of a ACA-based system, even while it is operating. In biological terms we would say that an organism continues its usual biological functions, even while growing or repairing itself. In short, synchronous CA may be too limited to model systems in nature, and as a result we may miss interesting behavior by restricting ourselves to them.

Somewhat in between synchronous and asynchronous CA are Self-Timed Cellular Automata (STCA) [12, 13]. In STCA a state transition of a cell only takes place as reaction to signal(s) the cell receives from neighboring cell(s). Rather than having a global mechanism synchronizing the cells, like in a synchronous CA, the cells in an STCA are locally timed, i.e., they trigger each other's operations. The workings of such systems resemble chain reactions in nature. Self-timing allows a better control of the operations of cells and allows for modular structures of systems [14]. The ease of controllability of the timing of synchronous CA is thus combined with the asynchronous operation of the ACA model.

The STCA concept put forward in [12, 13] has some restrictions such as the inability of triggering a cell's transition by more than one neighbor; the present paper puts the STCA concept into a more general and mathematically cleaner form, and shows transition rules for some functions that are often used in CA, like wires, crossings, and AND-gates.

The next section contains a description of STCA, followed by two sections on respectively triggering a cell's transition from a single neighbor, and triggering it from multiple neighbors. This paper finishes with conclusions.

2 The Self-Timed Cellular Automaton Model

A Self-Timed Cellular Automaton (STCA) consists of an array of cells, each of which has a finite neighborhood of, say, n cells. This paper deals only with 2-dimensional CA in which each cell has a neighborhood consisting of the four non-diagonal neighboring cells (von Neumann neighborhood). However, this can easily be generalized to other dimensions. A cell's state is made up of the actual state, which is a member of a finite state set Q, and a set of n "bit" states, each bit state being a member of the set $B = \{0, 1\}$, and corresponding to a unique neighbor of the cell. Each cell undergoes transitions in accordance to a transition rule that is a partial function:

$$f : Q^{n+1} \times B^n \times B^n \rightarrow Q \times B^n \times B^n.$$

The transition rule is defined as follows. Let q, q_N, q_E, q_S, q_W be the states of respectively a cell c, its north, east, south, and west neighbors, and b_N, b_E, b_S, b_W be the bit states of cell c that correspond to the north, east, south, and west neighbors respectively. Also, let a_N, a_E, a_S, a_W be respectively the southern bit state of the northern neighbor, the western bit state of the eastern neighbor, the northern bit state of the southern neighbor, and the eastern bit state of the western neighbor. The bits a_N, a_E, a_S, and a_W will be called the bits *adjacent to c*. The above symbols with primes attached to them denote the respective states after the transition has taken place.

Then the rule in accordance to which a cell undergoes transitions is given by:

$$f(q, q_N, q_E, q_S, q_W, a_N, a_E, a_S, a_W, b_N, b_E, b_S, b_W) = (q', a'_N, a'_E, a'_S, a'_W, b'_N, b'_E, b'_S, b'_W).$$

Fig. 1 shows the resulting transition graphically. The

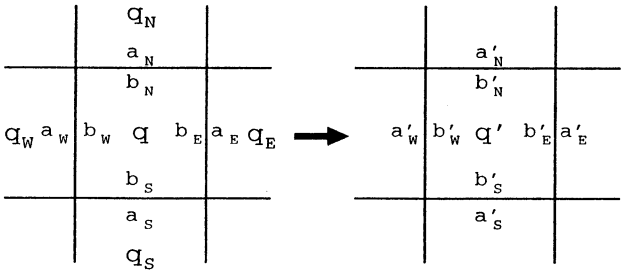


Figure 1: Transition in accordance to the rule f.

actual states of cells are depicted in the centers of the cells, and the bit states of the cells are depicted near the cells' borders with their neighbors. A bit with the value 1 is denoted by a filled circle, and a bit with the value 0 by an open circle. The STCA has no central clock signal: transitions of the cells take place asynchrously. Only if a transition rule is defined for the particular combination of states and bit states of a cell and its neighbors, will the transition take place. A transition can not be fired as long as it is undefined for a particular combination of states and bit states.

The following interpretation can be attached to bit states. A 1-bit of a cell c indicates that the cell *sends a signal* to the neighbor $\hat{c}$ corresponding to the bit. This signal may trigger the neighbor $\hat{c}$ to do a transition, which *may* return c's bit to 0. *May*—because it is not a necessity as in [12, 13], but in most cases it will be desirable to design transitions such that bits that trigger a cell's transition are returned to 0 by the transition. It is then said that the cell *acknowledges* the signals it received from the neighbors. Also, it is usually recommendable that the bits b_N, b_E, b_S, and b_W in the left hand side of a transition rule are 0. This means that no signals of the cell are pending to neighboring cells, and the cell is free to undergo a transition. However, in some special cases one may desire different behavior, for example to avoid deadlock, etc. This possibility is allowed by the STCA formalism (but not in the formalism defined in [12, 13]).

Fig. 2 shows a cell which receives a signal from its western neighbor, i.e., the neighbor's eastern bit is 1. This causes the cell to undergo a transition, in the

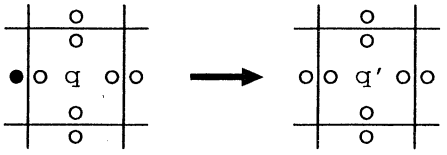


Figure 2: Cell receives a signal from its western neighbor (left); The cell acknowledges the signal, clearing the corresponding bit of the neighbor (right).

process acknowledging the signal by clearing the eastern bit of the neighbor. In this example, none of the cell's bit states is set to 1 by the transition. Section 3 deals in more detail with transitions in which cells are triggered by a 1-bit from a single neighbor.

The STCA formalism in this paper also allows for the triggering of a cell by more than one bit. For

example for the transition in Fig. 3 two bits of a cell's neighbors need be set to the value 1 to trigger it, and as a result of the transition one bit of the cell itself is set to 1. If only one of the bits of the two neighbors

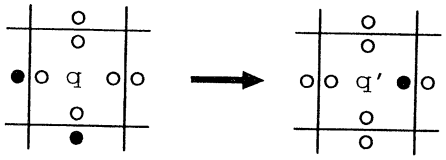


Figure 3: Two bits set to 1 trigger a cell, resulting, after transition, in their reset and the setting of a new state and bit state in the cell.

is 1, no transition will take place, unless a transition rule has been defined for that situation. Section 4 deals in more detail with transitions in which cells are triggered by 1-bits from more than one neighbor.

3 Single-Bit Triggering

This section gives an example of cells' operations triggered by a single 1-bit from the neighbors. The example concerns a wire that can transfer binary numbers to all directions other than the direction from which it receives a signal. It is implemented by cells having states denoted by $*_0$ and $*_1$, the 0- or 1-index indicating the number carried by the cell (this convention will also be used in the remainder of this paper). When the state of a neighbor does not make a difference as to the result, it is denoted by an underscore. Used as an index, an underscore indicates that any number may be carried by the cell.

A transition in accordance to the rule implementing this functionality is shown in Fig. 4. The transition

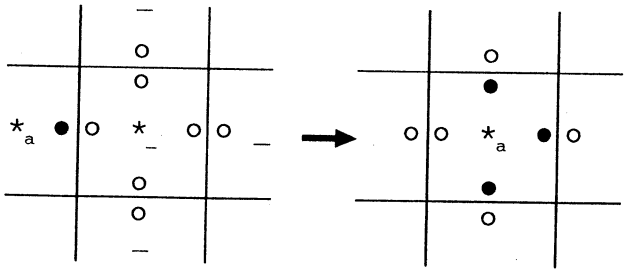


Figure 4: Transition of a wire cell. The value of a is 0 or 1.

rule gives rise to the following actions: when a cell receives a signal from its western neighbor, it copies the state of its western neighbor, and sends signals to the other three neighbors. Figure 5 shows the working of this rule on a wire configuration carrying the sequence of numbers $\langle 1, 0 \rangle$. The timing of signals is nondeter-

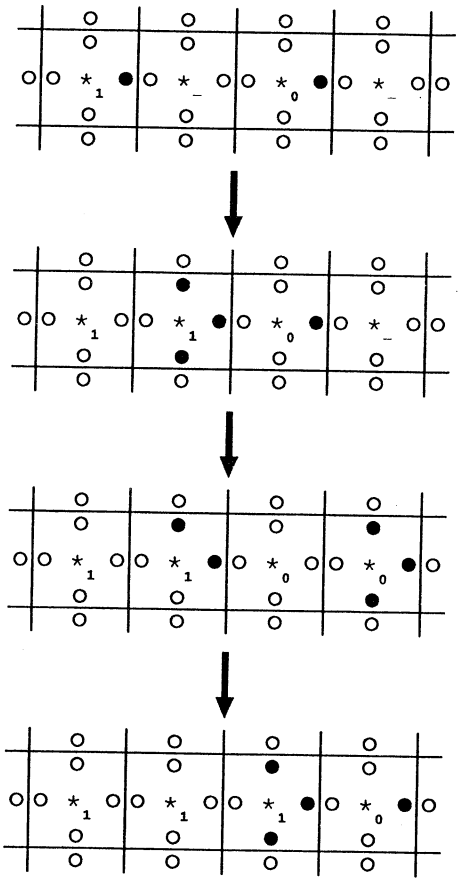


Figure 5: Transmission of the sequence $\langle 1, 0 \rangle$ along a wire. The index of a cell's state indicates the number carried by the cell. An underscore in the index indicates that any number may be carried by the cell.

ministic, giving an arbitrarily narrowing or widening gap between consecutive active cells at times. The ordering on a wire of two consecutive numbers in the sequence, however, will never change: a number can not surpass its successor in the sequence, because a cell will only process incoming signals when all its outgoing signals are acknowledged (by having the corresponding bits being reset to 0 by the neighbors that received the outgoing signals). The transmission rules for wire cells receiving signals from their northern, eastern, or southern neighbors are defined in a similar way as in this example.

To create circuits of wires, wire cells are used in conjunction with *quiescent* cells, which are cells that ignore any signals from their neighbors and of which the state is denoted by a blank (see Fig. 6 for a typical transition). Crossings of wires are not possible

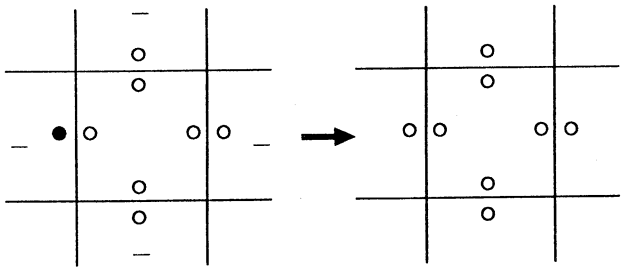


Figure 6: Transition of a quiescent cell.

by wire cells: for this we need so-called *crossing-wire cells*, which are described in the next section.

4 Multi-Bit Triggering

This section illustrates by some examples the operation of cells in case they receive signals from more than one neighbor. We start with a type of cell that needs signals from two neighbors in order to be triggered. The cell has states A_0 and A_1 and it acts as an AND-gate on the input it receives from the two neighbors. Fig. 7 shows a typical transition of this type of cell. If only one neighbor of an AND-gate cell sends a signal, the signal keeps pending and the cell will not be triggered until a signal from the second neighbor arrives. Once it arrives the two inputs are ANDed and the cell sends signals to its remaining two neighbors. The transition is thus only applied when both input bit states are 1.

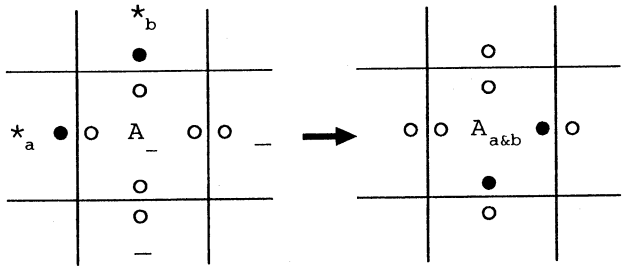


Figure 7: Transition of an AND-gate cell. The values of the indices a and b can be 0 or 1.

Finally, we describe a type of cell for which the situation is more complicated: the crossing-wire cell. This type of cell allows signals to cross each other without interference. The cell can be in the states denoted by the symbols $+_0$ and $+_1$. Operation of the crossing-wire cell is in accordance to the transitions in Fig. 8. The transitions can be separated into two types: cells that receive inputs from two neighbors, and cells that receive inputs from one neighbor. The first case covers the first transition in Fig. 8. Here, the signal coming from the west is processed first and once it disappears to the east the way is free for the second signal, from the north, to be processed in accordance to the second transition in Fig. 8. If there is only one signal from the west in the first place, the third transition in Fig. 8 will be used to cross the wire. In conclusion, the crossing-wire cell may be triggered by a signal from a single neighbor, but also by signals from two neighbors.

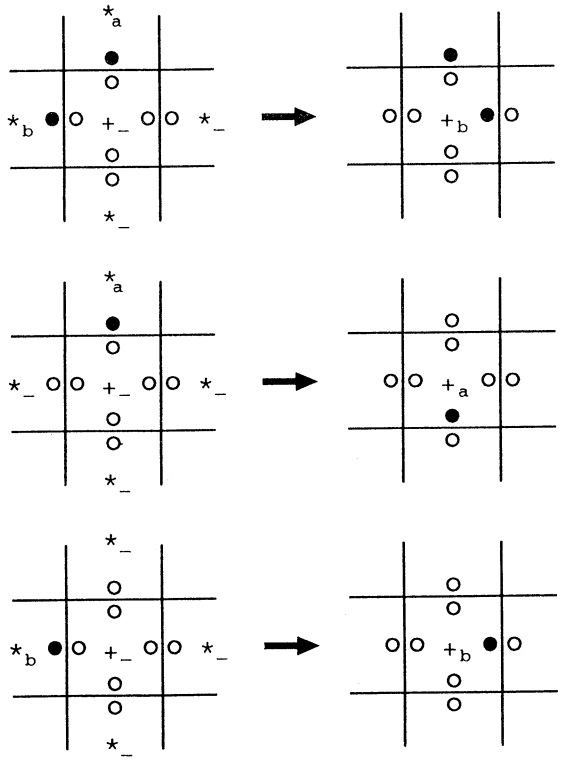


Figure 8: Possible transitions of a crossing-wire cell. The values of the indices a and b can be 0 or 1.

5 Conclusions

This paper gives a definition of STCA that is more general than the definition in [12, 13], which only covers single-bit triggering and which uses distinct mechanisms for signaling between cells and for transitions. The STCA definition in this paper unifies these mechanisms, and in addition has more expressive power. It allows nonstandard signaling behavior, not allowed in [12, 13], that may avoid certain anomalies, such as deadlock, etc.

Compared to conventional CA, the STCA transition mechanism lacks strict locality of action, in the sense that a transition not only changes the state and bit states of a cell itself, but also the adjacent bit states. This lack of locality, however, is very restricted: every bit state can only be changed by transitions of two cells, i.e., the cell to which the bit belongs and the cell to which the bit is adjacent. Furthermore, in standard signaling, the cell to which the bit belongs is only allowed to set the bit to 1, whereas the cell to which the bit is adjacent is only allowed to reset it to 0. These actions take place at distinct times, ruling out conflicting access to the bit.

From a biological point of view, the STCA model may be helpful for simulating systems in nature, since it incorporates a mechanism that is widely found in nature: chain reactions. This mechanism is found on the molecular level in chemical reactions, but also on the cellular level. For example, the signaling and transitional behavior of cells in STCA resembles the spiking behavior of neural cells in animals. In fact, simple neurons can be simulated by STCA when we assume that bit states of value 1 act as spikes and that they, when input to a cell, add up to the value of the cell's state, which is defined as an integer in a certain range, say $[0, n\rangle$. Once the state value of a cell would exceed the value n as a result of the input signals, it is returned to 0 and the cell's bit state at its output is set to 1. In neural network terminology, we have a neuron whose action potential is increased by input signals, until a certain threshold n is reached, at which the neuron produces an output signal and returns its action potential to 0. Implementation of other biological functions on the STCA model, such as evolution, self-reproduction, etc., can also be imagined.

Apart from their biological plausibility, STCA also offer much promise as the underlying model for computers realized by future technologies, like nanotechnology. As pointed out in [14], programs for such computers would be build up of hierarchically organized modules that are layed out spatially over cellular space and that would interact in a self-timed way. The self-timed characteristics of the modules would allow designers of modules to focus on the functionality of modules rather than on timing of signals in modules, the latter being a difficult aspect in synchronous systems design [15, 16]. Once designed, modules could be fit seamlessly together, again without any worry about timing matters. Furthermore, self-timedness of underlying hardware (e.g. molecular electronics) would allow only those elements to be active that are actually operating, as opposed to synchronous systems, where all elements are active every clock cycle. This would substantially reduce the power consumption of such devices.

We conclude that STCA are a promising model, not only for simulating biological systems, but also as the underlying framework on which future models of computation may be based.

Acknowledgements

This research is financed by the Japan Ministry of Posts and Telecommunications as part of their Breakthrough 21 research project.

References

[1] E.F.Codd, "Cellular Automata", New York: Academic Press, 1968.

[2] J.von Neumann, Ed: A.W.Burks, "Theory of Self-Reproducing Automata", University of Illinois Press, 1966.

[3] K.Morita and K.Imai, "Self-reproduction in a reversible cellular space", Theoretical Computer Science, Vol. 168, 1996, pp. 337-366.

[4] M.Markus, A.Czajka, D.Böhm, et al., "Phenomenology of Cellular Automata Simulations of Natural Processes", In: E.Goles and S.Martínez (eds.), Cellular Automata and Complex Systems, 1999, Kluwer Academic Publ., pp. 55-105.

[5] S.Wolfram, "Universality and Complexity in Cellular Automata", Physica D, Vol. 10, Jan. 1984, pp. 1-35.

[6] R.Serra, M.Villani, and A.Salvemini, "Genetic Network Models of Biodegradation", In: "Cellular Automata: Research Towards Industry, ACRI'98 Proc. 3rd Conf. on Cellular Automata for Research and Industry", Oct 1998, Trieste, Italy, pp. 203-217.

[7] S.Wolfram, "Cellular Automaton Fluids: Basic Theory", Journal of Statistical Physics, Vol. 45, Nov. 1986, pp. 471-526.

[8] S.Morishita and S.Itagaki, "Modeling of Dynamic Systems by Cellular Automata", CA 2000, Proc. Int. Workshop on Cellular Automata, 6th IFIP WG1.5 Meeting, 21-22 Aug 2000, Osaka, Japan, pp. 32-33.

[9] L.Priese, "Normed Networks: Their Mathematical Theory and Applicability", In: "Applied General Systems Research", ed. G.Klir, Nato Series, Series II: Systems Sciences, 1978, pp. 381-394.

[10] K.Nakamura, "Asynchronous Cellular Automata and their Computational Ability", Systems, Computers, Controls, Vol. 5, No. 5, 1974, pp. 58-66 (Translated from Denshi Tsushin Gakkai Ronbunshi, Vol. 57-D, No. 10, Oct. 1974, pp. 573-580.)

[11] T.E.Ingerson and R.L.Buvel, "Structure in Asynchronous Cellular Automata", Physica D, 1984, Vol.10, pp. 59-68.

[12] T.Isokawa, F.Peper, N.Kouda, et al., "Computing by Self-Timed Cellular Automata", Proc. of the 6th IFIP International Workshop on Cellular Automata, 21-22 Aug. 2000, pp. 3-4.

[13] F.Peper, T.Isokawa, N.Kouda, et al., "Self-Timed Cellular Automata", Proc. of the Society of Instrument and Control Engineers, Symp. of the Kansai Chapter, 4-5 Oct. 2000, pp. 154-157.

[14] F.Peper, "Spatial Computing on Self-Timed Cellular Automata", UMC'2K, Proc. 2nd Int. Conf. on Unconventional Models of Computation, 13-16 Dec. 2000, Brussels, Belgium.

[15] R.M.Keller, "Towards a Theory of Universal Speed-Independent Modules", IEEE Trans. on Comp., Vol. C-23, No. 1, Jan. 1974, pp. 21-33.

[16] I.E.Sutherland, "Micropipelines", Communications of the ACM, Vol. 32, No. 6, 1989, pp. 720-738.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

OPTICALLY PROGRAMMABLE COMPUTATIONS WITH DNA MOLECULES

Danny van Noort and John S. McCaskill

BioMolecular Information Processing, GMD, 53764 Sankt Augustin, Germany
tel.: +49 2241 14 1521/1526, *fax:* +49 2241 141511
email: danny.van-noort@gmd.de; mccaskill@gmd.de, *webpage:* http://www.gmd.de/BIOMIP

ABSTRACT

The goal of this research is to improve the programmability of DNA based computers. Novel clockable micro-reactors can be connected in various ways to solve combinatorial optimisation problems, such as Maximum Clique and 3-SAT. This work demonstrates that one micro-reactor design can be programmed optically to solve any instance of Maximum Clique up to its given maximum size (N). It reports on an implementation of the concept proposed previously [3]. The advantage of this design is that it is generically programmable. This contrasts with conventional DNA computing where the individual sequence of biochemical operations depends on the specific problem. Presently we are solving a graph for Maximum Clique with N=6 nodes and have completed the design of a microreactor for N=20. We are planning to extend this research with re-configurable microflow reactors to evolutionary microfluidic bio-electronic systems.

Keywords: *DNA, microflow reactor, optical programmability, reconfigurable, maximum clique, evolutionary computing.*

INTRODUCTION

DNA computing involves a multidisciplinary interplay between molecular biology, microsystem technology, physical detection and evolution. Since the introduction in 1994 by Adleman [1] there has been intensive research into the use of DNA molecules as a tool for calculations, simulating the digital information processing procedures in conventional computers. In the short term, however, the main application of DNA computing technology will be rather to perform complex molecular constructions, diagnostics and evolutionary tasks. However, in order to assess the limits of this technology, we are investigating a benchmark computational problem: Maximum Clique, chosen as an NP-complete problem because of its limited input information [2]. In our research we took the step from batch processing in test tubes to pipelined processing in micro-flow reactor networks [3]. This gives us complete control over the process of information flow and allows operations much faster than in conventional systems. More importantly, it allows the extension to optical programming. Moreover, the proposal differs radically from the surface based DNA computing approach [4] in requiring no problem dependent manual or robotic operations, programming being completely under light control.

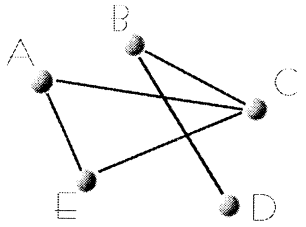


Figure 1: *a 5-node representation of the clique problem. The maximum clique is given by* ACE *represented by 10101.*

MAXIMUM CLIQUE

The decision problem associated with the maximum clique problem is a hard to solve NP-complete graph problem. Maximum clique requires finding the largest subset of fully interconnected nodes in the given graph (Fig. 1). To obtain the set of cliques and then determine its largest member using a micro-flow system, an algorithm was devised consisting of a series of selection procedures containing three parallel selection steps.

The problem can be divided into two parts: (i) find all the subsets of nodes, which correspond to cliques in the graph and (ii) find the largest one. The basic algorithm is simple [2]: for each node i ($i \geq 1$) in the graph retain only subsets either not containing node i or having only other nodes j such that the edges (i,j) are in the graph. This can be implemented in two nested loops (over i and j), each step involving two selectors in parallel.

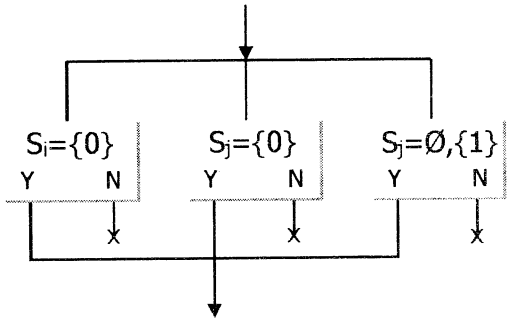


Figure 2: *a flow diagram showing the selection procedure for node i as whether it is connected to node j and is part of the cliques.*

A third selector has been introduced to allow the selector sequences to be fixed independently of the graph instance. Thus the graph dependence is programmed not by which but by whether a sub-sequence selection in the third selector is performed (see Fig. 2). The above procedure is described in more detail by McCaskill [3].

The connectivity matrix for the 5-node example shown in Figure 1 is the 5x5 matrix in Table 1.

Table 1: *the connectivity matrix for the 5-node graph as shown in Figure 1.*

	A	B	C	D	E
A	1	0	1	0	1
B	0	1	1	1	0
C	1	1	1	0	1
D	0	1	0	1	0
E	1	0	1	0	1

As Table 1 shows, the matrix is symmetrical over the diagonal, reducing the number of selections from N^2 to ½N(N-1).

SELECTION MODULE

The above procedure can be implemented into a network of micro-reactors. To this end we have developed a module which is able to make positive sub-sequence selections from a population of specific DNA sequences. Each DNA sequence encodes a binary sequence corresponding to a particular subset of nodes in the graph. Different DNA sub-sequences are used to represent presence (1) or absence (0) at each node.

To transfer the selected DNA sequences to the appropriate output, they are transferred from one flow to another by moving paramagnetic beads, on which single stranded DNA complementary to a nodal sub-sequence is immobilised. Amino labelled selector strands are photo-immobilised to carboxyl-functionalised beads (silicon, latex or glass), under the control of light [5]. The DNA strands in solution hybridise to the selector strands and are thus transferred to an other channel in the micro-flow reactor where they are de-hybridised and passed on to the next selection procedure (see Fig. 3).

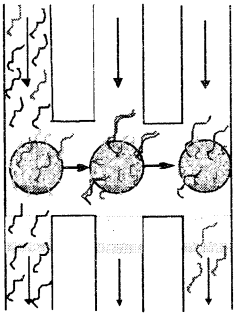


Figure 3. *The Selection Transfer Module (STM). The DNA template comes in through channel A. Some of the DNA-strands will hybridise to the selection strand immobilised to the paramagnetic bead. This bead is moved through a wash channel B, to rinse off the unbound strands, into the de-hybridisation channel C from where the selected strands will continue, through a neutralisation step, to the next STM.*

The STMs have three supply channels, for the solution template, the washing solution and de-hybridisation buffer respectively. De-hybridisation is performed by using an alkali solution (NaOH), adjusted for the common "melting" temperature of the hybridised DNA-strands. Because of the change in pH at the de-hybridisation step, a subsequent neutralisation step is necessary (see Fig. 4).

The STMs can be connected in various ways to solve combinatorial optimisation problems, such as Maximum Clique and 3-SAT. The whole micro-reactor configuration is photo-lithographically etched on 4" silicon substrates. The etched wafer is sealed with an anodically bonded pyrex glass wafer. Capillary tubing is attached through ultrasound drilled holes in the pyrex wafer.

PROGRAMMABILITY

These Strand Transfer Modules (STM) can be optically programmed as outlined in [3], by means of photo-immobilisation [5], thus creating a re-configurable micro-fluidic computer. Unlabeled beads are delivered in parallel at the appropriate locations to each STM. During the initial set-up

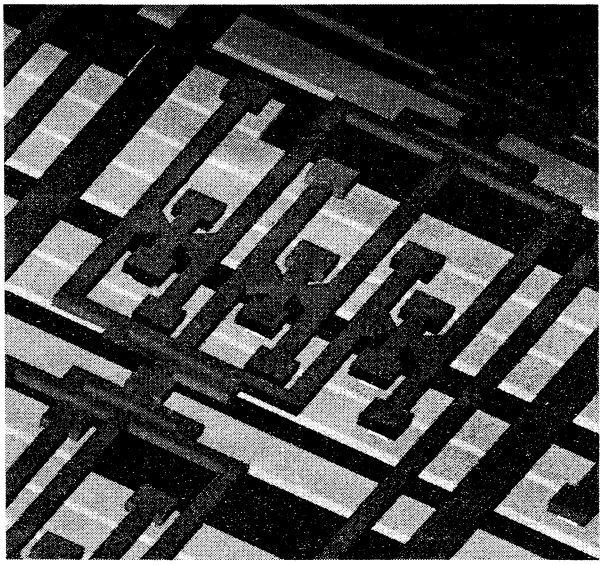


Figure 4. *The 3 selection modules for a node connectivity decision (top layer) with the supply and waste channels (bottom layer).*

phase, only one type of selection strand flows past any given bead. UV laser light then determines whether or not a sequence is immobilised on the beads at each STM. If there is no connectivity between a pair of nodes, the corresponding third STM is not loaded. The other STMs can be preloaded with immobilised beads since they will never have to be changed when re-configuring the problem. The immobilisation pattern is directly related to the connectivity matrix.

The information flow can be tracked using a sensitive CCD detection system to detect laser-induced fluorescence with intercalating dyes or labelled DNA Because of the fluorescent information from each STM with a correct DNA-strand transfer, it is possible to monitor the solution of the algorithm to its conclusion over time. The STMs can also be used for selecting subsets with a given number of nodes, and in the implementation below this separation procedure follows the clique selection (see Fig. 6).

TECHNOLOGY

To set up a DNA-computer in a micro-flow system puts high demands on the control system, an overview of which is presented in Figure 5.

To distribute the DNA-template and buffer solutions to the wafer a liquid-handling system is connected. It consists of a pipetting robot and a series of multi-position valves which control the solution distribution. One of the multi-position valves makes it possible to address the individual immobilisation sites in the STMs for the selection strand for a more convenient serial setup phase. To programme the computer optically a microscope is used to project laser light from an EXCIMER laser (308 nm), either serially or through a static (later dynamic) mask. The wafer is mounted on an xy-translation stage so as to address all the STMs individually. A detection system consisting of a CCD camera for a general overview and a microscope for detailed pictures is in place. The clique and subset size selection microflow reactor is shown in Fig. 6.

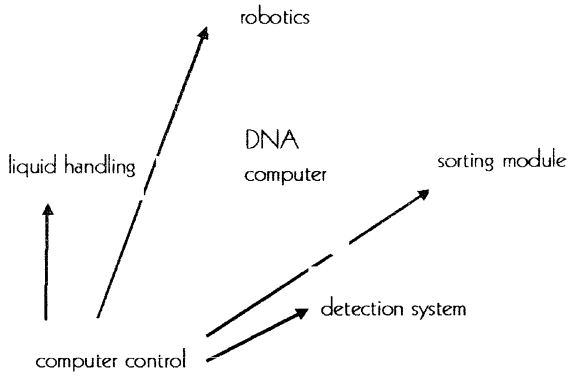


Figure 5. *An overview of the operation of a configurable DNA computer.*

The beads are moved by a magnet which sweeps over the DNA-computer's surface. This sweep clocks the serial steps of the computation, which may be pipelined to increase throughput

Presently we have constructed microflow reactors for N=6 and N=20 nodes and are testing performance firstly with the N=6 version requiring 15 (N(N-1)/2) STMs and 15 (2N+3) inlets. The number of inlets scales linearly and the number of modules quadratically in the problem size.

FUTURE

A further way to increase the flexibility of the programmable DNA-computer is to construct microflow reactors with re-configurable connectivity. The configuration can be externally changed by redirecting fluidic routes between reactors, mixers and detectors in analogue with Field Programmable Gate Arrays [6]. The advantages here are that any program can be implemented without reconstructing the wafer and the number of microreactors can be limited. By coupling them to biomechanical systems and electronic hardware components, complex construction systems can be achieved which are evolvable on different hardware levels [6].

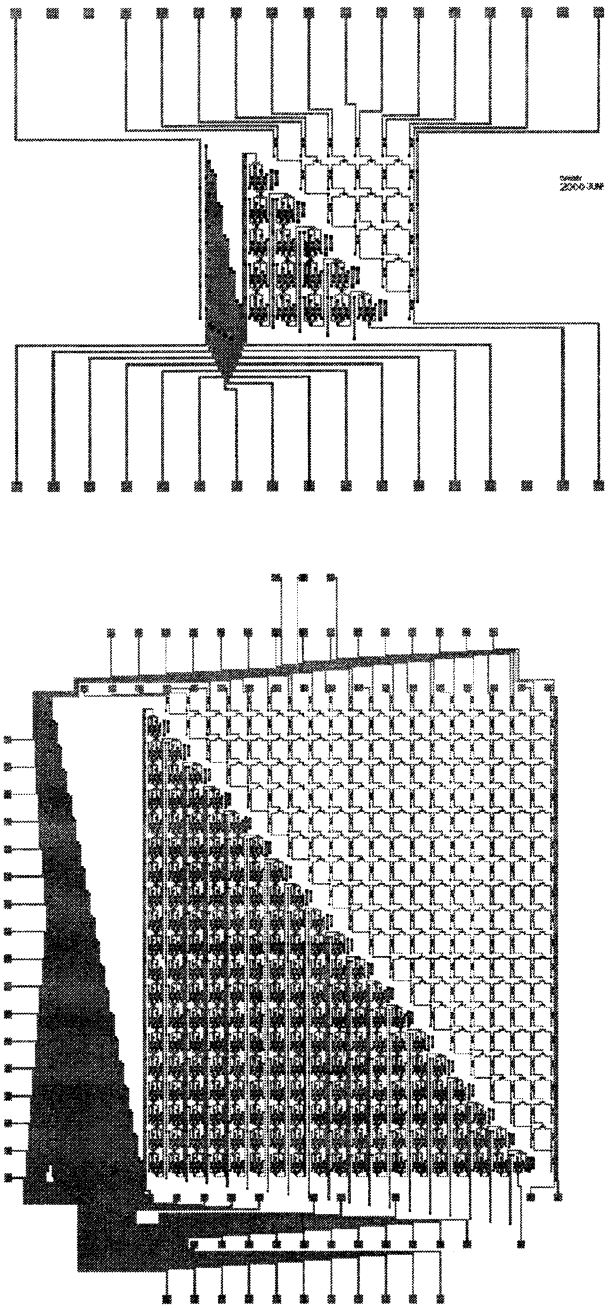


Figure 6. *The complete top layer of the DNA-computer for a 6- and 20 node graph. The top right triangle does the length selection, while the other half does the clique selection. The square pads are the input and outputs of the DNA-computer, while the rest of the channels are supply channels. These are connected on the back of the wafer with the reactors, as shown in Figure 4.*

In the more distant future we can envisage evolving DNA-computers in which the programme writes itself to optimise the microflow configuration and to adapt to failures in the computer.

ACKNOWLEDGEMENT

The authors wish to acknowledge the support of the GMD and of the group's start-up grant (#01SF9952) from the German Ministry of Science (BMBF).

We would like to thank Patrick Wagler, Marlies Gohlke, Robert Penchovsky, Jörg Ackermann and Harald Mathis for their assistance with the microreactor implementation and set-up.

REFERENCES

[1] Adleman, L. M. (1994) *Molecular computation of solutions to combinatorial problems*. Science **266**, 1021-1024.

[2] Ouyang, Q., Kaplan, P. D., Liu, S. and Libacher, A. (1997) *DNA solution to the maximal clique problem*. Science **278**, 446-449.

[3] McCaskill, J. S. (2000) *Optically programming DNA computing in microflow reactors.* Submitted.

[4] Liu, Q., Wang, L., Frutos, A. G., Condon, A. E., Corn, R. M. and Smith, L. M. (2000) *DNA computing on surfaces.* Nature **403**, 175-179.

[5] Penchovsky, R., Birch-Hirschfeld, E. and McCaskill, J. S. (2000) *End-specific covalent photo-dependent immobilisation of synthetic DNA to paramagnetic beads*. Nucleic Acids Research **28**, e98, 1-6.

[6] McCaskill, J. S. and Wagler, P. (2000) *From reconfigurability to evolution in construction systems: spanning the electronic, microfuidic and biomolecular domains.* **In** R. W. Hartenstein and H. Grünbacher (Eds.) FPL 2000, LNCS 1896, pp. 286-299, Springer-Verlag, Berlin Heidelberg.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A Novel Class of Super-Turing Machines and Their Robustness Analysis

Jian-Qin LIU[1,2] and Katsunori SHIMOHARA[1]

[1] ATR International, Information Sciences Division
Soraku-gun, Seika-cho, Hikaridai 2-2, Kyoto, 619-0288, JAPAN
E-mail: {jqliu, katsu}@isd.atr.co.jp

[2] College of Information Eng., Central South Univ., CHINA

Abstract -- This paper proposes a novel computational model based on proteomic computing leading to the construction of super-Turing machines. The dynamical description for pathways, the evolutionary mechanism and robustness are discussed. Furthermore, a preliminary simulation experiment shows the merits of our method for potential applications.
Keywords: Super-Turing Machines, Proteomic Computing, Robustness, Evolutionary Moleware, Molecular Electronics.

1 Introduction

In order to improve the performance of computers, new paradigms continue to emerge, e.g., DNA computing, quantum computing and molecular electronics. Considering the limits of conventional computers in the aspects of computability and speed, active efforts for exploring new architectures for next-generation computers have been made in terms of new media for implementation such as molecules or quantum devices. The implementation of DNA computing is closely related to bio-molecules [1~8], where evolutionary wetware [7,8] can be applied (e.g., for code designing). Quantum devices and potential fast computation by molecular electronics [9~13] are also receiving attention. As a new kind of molecular computing, "proteomic computing" has been proposed [14,15]. In the view of theoretical computer science, the performance of abstract machines is complicated at the boundary of decidability and undecidability, e.g., "emergeablility" for certain functions of evolutionary computing. Hava T. Siegelmann's work [16] on super-Turing machines merits attention. The question of how to go beyond the limits of Turing machines and Gödel's incompleteness is our starting point from the theoretical aspect. Universal computability and undecidability are two important factors in the journey to explore super-Turing machines. The reasons can be found in the following three major points: (1) The "coupling" between the universal computation and undecidability. (2) The crucial parameters in unconventional computational mechanisms. (3) The complexity in a dynamic landscape derived from the underlying problem spaces. We are also trying to work out certain "operable" models for this target.

2 Proteomic Computing

According to the definition of "proteomic computing", the underlying computational processes will be analog (it can also offer computing by a super-Turing machine [14]). Further description is necessary for the mechanism concerned. Let A(0) be the input set (vector) and A(n) be the output set (vector).

i.e.

$$A(0) = \{A(0,0), A(0,1), \ldots, A(0,m_0\}$$

$$A(1) = \{A(1,0), A(1,1), \ldots, A(1,m_1\}$$

$$\ldots$$

$$A(n) = \{A(n,0), A(n,1), \ldots, A(n,m_n\}$$

where A(i,j) refers to the *jth* element in A(i). The relation between A(i) and A(i+1) (i = 0,1,..., n) is quantitatively represented by ODE (Ordinary Differential Equation). So we can get

$$d(A) / dt = W(i) * A(i)$$

where W(i) refers to the coefficient in the corresponding ODE and determines the interactions by zero or non-zero valuing as well. An explanation of the above processes can be found in [15] briefly and in [14] in more detail. All of the values for W(i) belong to R, making the searching space into a kind of whole space consisting of all of the situations leading to NP problem solving processes, where we introduce an evolutionary dynamics model and dynamical evolutionary algorithm derived casually. Fig.1 shows the evolutionary unit for proteomic computing. Compared to the commonly-used ES (Evolutionary Strategy), our proposed dynamical evolutionary algorithm is arranged as

$$X' = X + f(a, b, c, d)$$

where f (a, b, c, d) is generated by spatial-temporal sampling from deterministic chaos with the parameter set {a, b, c, d} in f (a, b, c, d) function, in which

a ---- determines the kinds of chaotic forms, e.g., strange attractor.

b ---- is the limitation for the valuing,

c ---- is the spatial sampling, and

d ---- is the temporal sampling.

The parameters such as a, b, c, and d are updated based on feedback from the dynamical environment, e.g., the circling ("survival") of the signal pathways in the meaning of artificial life. Here the main difference between the dynamical evolutionary computation and common ES is that f (.) is defined as a different functional form, mentioned above, from the stochastic ways, e.g., normal distribution.

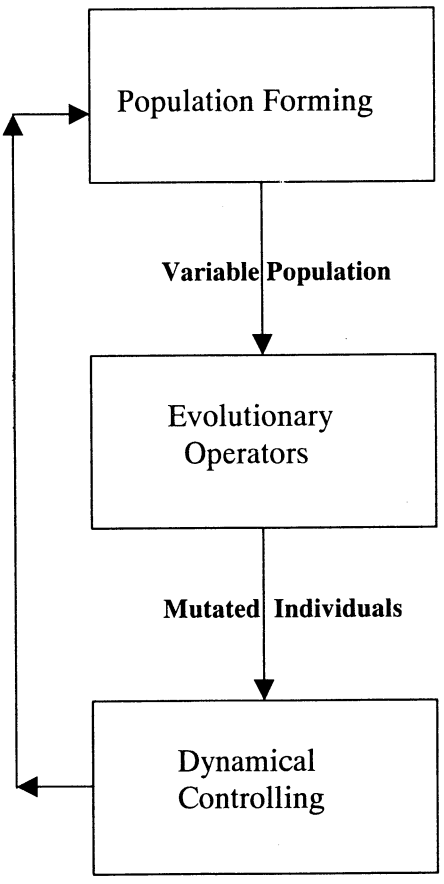


Fig.1 Controlling the Evolution

3 Robustness Mechanism

Artificial chemistry offers an alternative way to explore life compared with (conventional) biology. The methods vary at different levels [17]. Among them, DNA computation is regarded as an analogous modeling paradigm, as Dittrich points out in his tutorial ECAL99 & ALIFE 7 [17,18]. The discipline of artificial chemistry also covers more different and detailed methods (e.g. in ALIFE 7 [19], the methods by Alan Dorin [19] pp. 13-20, Hugues Bersini [19] pp. 39-48, Pietro Speroni di Fenizio [19] pp. 49-53, Yasuhiro Susuki and Hiroshi Tanaka [19] pp. 54-63, and Harald Hüning [19] pp. 64-72). In cell biology, signal transduction plays an important role in cells [20]. The problem we want to solve in our work for exploring robustness, is how to allow an evolutionary system to be embedded with a robustness mechanism that can enable the system to evolve a certain functional pathway to keep the cell alive where/when it faces a complex cellular environment full of uncertainty with respect to the dynamical organization for homeostasis. "Robustness is a central issue in all complex systems. For example, robustness is critical to self-assembling or self-repairing systems which subject to external perturbations, fluctuations, and noise. ..." [21]. Robustness here is defined as the features and the corresponding mechanism that can guarantee that the underlying systems will keep their expected performance under the condition that disturbances are allowed. Here the model of proteomic computing is based on a fine-grained representation for inter-cellular biochemical reactions. When certain parameters' set C are disturbed into C', the performance of the model can be kept within the domain in given requirements (e.g., maintaining the cycling of pathways through proper feedback):

$$M(H(C')-H(C)) < Tm$$

where M(.) is the necessary measurement for robustness in model H(.) and Tm is the threshold concerned.

4 Experimental Result

Fig. 2 (a) and (b) show the robustness of the proteomic computational model. Series 1 is the performance curve in normal situations. Series 2 is the one when the parameters have been disturbed into two values but the performance was remained stable (the variation in the two different parameters do not influence the performance curve) in abnormal situations.

5 Conclusion

We have initiated a new paradigm of molecular computing ---- "proteomic computing" leading to the

potential construction of an analog super-Turing machine with possibly non-Gödel logic forms [7].
As we previously mentioned [7]:

"Besides ... mentioned above, logic is another important feature of computational machines, but Gödel's incompleteness constrains some functional factors when we try to construct formal systems. ...
Beyond the limits of Gödel's incompleteness is another task in our ... computing research."

The motivation of our work (final target) is to establish a systematic and rigorous theory for unconventional computational systems as that a class of unconventional computational mechanisms can be embedded into derive super-Turing machines. Our preliminary simulation experiments for proteomic computing show broad applications, such as fault/defect tolerant molecular computers based on organic/inorganic molecular electronics, robust "evolvable" multi-agent systems, non-parametric natural computing for multi-robots in dynamical environments with uncertainty, and artificial brains based on molecular electronics [9].

Acknowledgement:
The authors are thankful to Prof. Sugisaka, Prof. Tanaka, Prof. Kaibuchi, Prof. Hagiya, Prof. Furuhashi, Dr. Kuroda, Dr. Amano, Dr. Peter Dittrich, Prof. Kaibuchi's Lab. at NAIST, and Prof. Hagiya's Molecular Computing Research Group that includes members in Tokyo Univ. and in other institutions. J.-Q.Liu is thankful to Dr. Nishikawa, Prof. Rozenberg, Prof. Mori, Dr. Ara, Dr. Gheorghe Paun, Dr. Hava T. Siegelmann, Dr. H. Suzuki, Dr. Y. Suzuki, Dr. M. Arita, Dr. Hemmi, Dr. Peter Eggenberg, Dr. Takadama, Mr. Noberto Nawa Eiji and Assoc. Prof. Minjie Wei. The work is partly supported by Huo Yingdong Foundation (project no.71063).

References

[1] Adleman, L.M., Molecular computation of solutions to combinatorial problems, Science, 266, 1994; 1021-1024.

[2] Mitsunori Ogihara and Animesh Ray, DNA computing on a chip, Nature, 403; 143-144.

[3] Qinghua Liu, Liman Wang, Anthony G. Frutos, Anne E. Condon , Robert M. Corn and Lloyd M. Smith, DNA computing on surfaces, Nature, 403; 175-179.

[4] Masami Hagiya, Perspectives on molecular computing, New Generation Computing, 17, 1999; 131-151.

[5] Akio Nishikawa and Masami Hagiya, Towards a system for simulating DNA computing with whiplash PCR, Proc. of CEC'99, 1999; 960-966.

[6] Kensuku Sakamoto, Hidetaka Gouzu, Ken Komiya, Daisuke Kiya, Shigeyuki Yokoyama, Takashi Yokomori and Masami Hagiya, Molecular Computation by DNA Hairpin Formation, Science, Vol.288, 2000; 1223-1226.

[7] Jian-Qin Liu and Katsunori Shimohara, DNA computing by genomic dynamics I: evolutionary modeling of emergence and context-sensible grammar representation, [Eds. M.Sugisaka & H.Tanaka] Proc. of the Fifth International Symposium on Artificial Life and Robotics (Oita, Japan, Jan.26-28, 2000); 781-784.

[8] Jian-Qin Liu and Katsunori Shimohara, DNA computing by genomic dynamics II: a simulation wetware prototype of dynamical DNA computation, [Eds. M.Sugisaka & H.Tanaka] Proc. of the Fifth International Symposium on Artificial Life and Robotics (Oita, Japan, Jan.26-28, 2000); 785-788.

[9] Philip Ball, Chemistry meets computing, Nature, 406, 2000; 118-120.

[10] C.P. Collier, E.W. Wong, M. Belohradsy, et al., Electronically configurable molecular-based logic gates, Science, 285; 1999; 391-394.

[11] James R. Heath, Philip J. Kuekes, Gregory S. Snider, R. Stanley Williams, A defect-tolerant computer architecture opportunities for nanotechnology, Science, 280, 1998; 1716-1721.

[12] Mark A. Reed and James M. Tour, Computing with molecules, Scientific American, June 2000; 69-75.

[13] Thomas Rueckes, Kyoungha Kim, Ernesto Joselevich, et al., Carbon nanotube-based nonvolatile random access memory for molecular computing, Science 289, 2000; 94-97.

[14] Jian-Qin Liu and Katsunori Shimohara, Proteomic Computing: theory and practice, ATR Technical Report, 2000.

[15] Jian-Qin Liu and Katsunori Shimohara, Fine-grained Molecular Computers Based on Molecular Electronics, full paper in Journal of Three Dimensional Images (2000), abstract in Proceedings of the Third International Conference on Human and Computer (HC-2000), (September 6-9, 2000, Aizu-Wakamatsu, Japan), p.318.

[16] H.T. Siegelmann, Computation beyond the Turing limit, Science, 268 (5210), April 1995; 545-548.

[17] Peter Dittrich, Artificial Chemistries, Tutorial held at ECAL'99 (European Conference on Artificial Life, 13-17 September1999,Lausanne,CH), http://ls11-www.informatik.unidortmund.de/achem/tutorialAChemECAL99-normal.ps.gz

[18] Peter Dittrich, Personal Communication, September, 2000.

[19] Mark A. Bedau, John S. McCaskill, Norman H. Packard, and Steen Rasmussen, (Eds.), Artificial Life VII (Proceedings of the Seventh International Conference on Artificial Life), A Bradford Book, The MIT Press (Cambridge, Massachusetts; London, England), 2000.

[20] K. Kaibuchi, S. Kuroda and M. Amano, Regulation of the cytoskeleton and cell adhesion by the Rho family CTPases in mammalian cells, Annu.Rev.Biochem. 68, 1999; 459-486.

[21]http://www.santafe.edu/sfi/reseach/focus/robustness/index.html

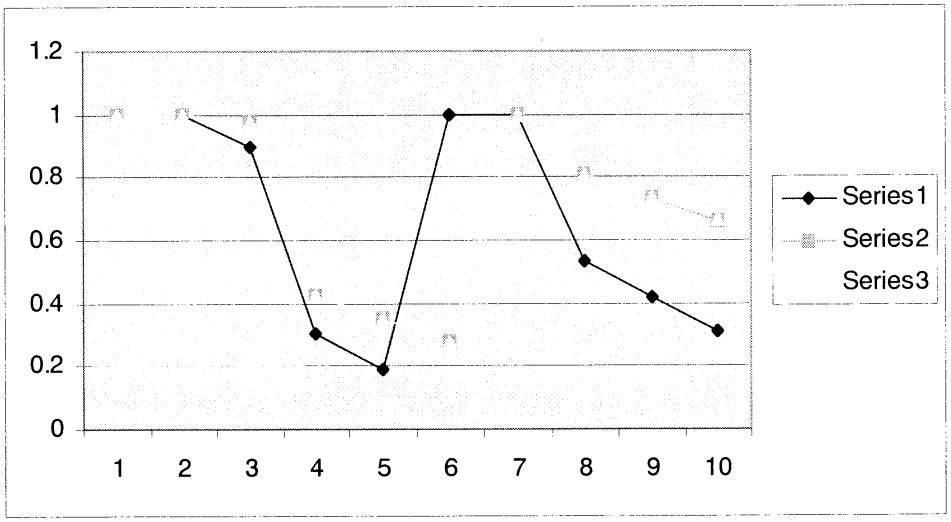


Fig. 2 (a) Activity vs. time

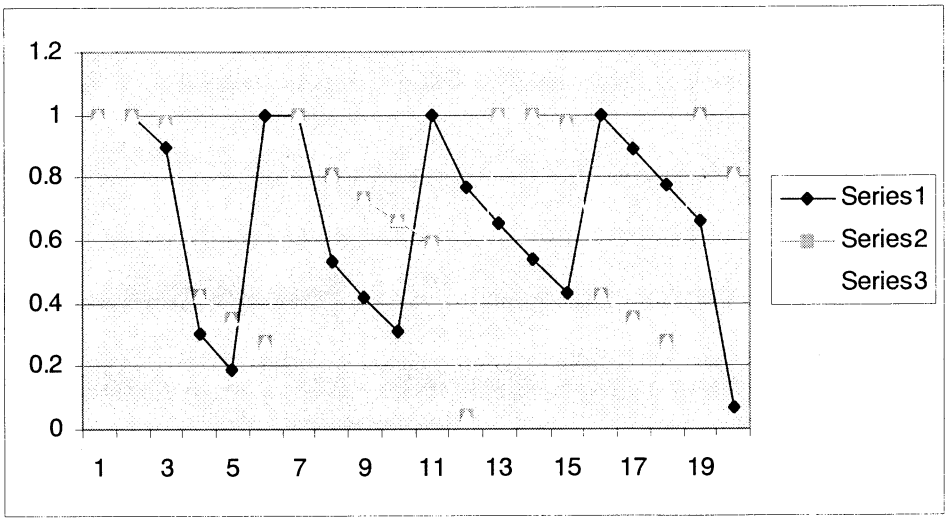


Fig.2 (b) Activity vs. time

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Symbiotic Artifact Design by Bio-informatic and Multimodal Framework

Osamu Katai*, Hiroshi Kawakami*, Hidetsugu Suto* and Kentaro Toda**

* Dept. of Systems Science, Graduate School of Informatics, Kyoto University
{katai/kawakami/suto}@sys.i.kyoto-u.ac.jp
** Dept. of Earth and Space Science, Graduate School of Science, Osaka University
Machikane-yama 1-3, Toyonaka, Osaka Pref. 560-0043, Japan

Abstract

Various novel notions such as Modal Design and Autoprescriptive Sets are introduced to extend our conceptualization of life system as multi-body symbiotic architecture of modules. Also, a new paradigm of anthropology called "Multimodal Framework (MMF)" introduced by G. Samuel is incorporated to our architecture. Modal Design is used to design these complex architectures, and autoprescriptive sets can be utilized for progressing the processes of modal design.

1 Introduction

In this paper, we will extend our notion of multi-body symbiotic system architecture of bio-informatic systems that focuses on the role of information on the highly complex and hierarchically organized structure of life systems.

For this purpose, we will introduce the newly born paradigm of anthropology called "Multimodal Framework (MMF)" introduced by G. Samuel [1]. He emphasizes on the way of conceptualizing culture as the interface between biology and anthropology. The basic notions of this framework, modal states and modal currents, are then related to our architecture.

For designing these complex multi-body systems, we will introduce a novel notion of design, "Modal Design", where the process of deign is appropriately confined to derive desirable solutions depending on the context and the progress of design.

Hence, the design problem is reduced to the "design of modal design", a kind of "meta-design". For solving this meta-problem, we will also introduce a novel notion of sets, "Autoprescriptive Sets". The way of deriving these sets and their properties are discussed.

2 The notion of Autoprescriptive Sets

We are interested in an novel notion of set concept, more precisely, a notion of class of sets, which we will call "Autoprescriptive Sets". This notion is not unusual in pure mathematics, in axiomatization of various systems such as Euclidean Geometry, where basic terms are not defined overtly, but are done in implicit ways.

Let us consider the case of defining a set C by referring to C itself. That is,

$$C = \{x|F(x;C)\}$$

where $F(x;C)$ is a predicate on x with reference to C. Let a set function G be defined as

$$G(D) = \{x|F(x;D)\}$$

Then, we have the following equation on C such as

$$C = G(C)$$

That is, C is given as a "fixed point" of the above equation. Thus, C may not be unique, and hence, it may better be regarded as a class (collection) of sets.

In the following, we will not examine the process of solving this equation such as computability and decision problems.

Let us consider a special case where G has monotonic properties as follows:

$$C \supset C' \Rightarrow G(C) \supset G(C')$$

And we will introduce the following formal power series.

$$D' = D \cap G(D) \cap G^2(D) \cap \cdots$$

Then we have

$$D' \supset G(D')$$

which thus leads to

$$D' \supset G(D') \supset G^2(D') \supset \cdots \supset G^n(D') \supset G^{n+1}(D') \supset \cdots$$

Hence, we have a fixed point solution

$$D^{\infty} = G(D^{\infty}),$$

when an appropriate topology is introduced on the lattice structure of sets.

We are interested in the stability and the instability of these fixed points. Particularly, we are interested in the relation between two different fixed points such as if they are disjoint or not.

Concerning this, the researches on Nonmonotonic Reasonings and Common-sense reasoning are quite interesting. For instance, D. McDermott and Doyle [2] introduced a fixed point solution (interpretation) of theorems (inferable propositions) of non-monotonic theories. Also, R. Reiter [3] introduced the notion of default reasoning and have shown that two different theories (set of theorems) are, if they exists, necessarily inconsistent with each other.

3 MMF: Multimodal Framework

G. Samuel [1] introduced by his book "Mind, body and culture: Anthropology and the biological interface" the interesting notion of "modal states" and "modal currents" to form the framework which he calls "multimodal framework (MMF)". MMF consists of three basic constituents, that is,

MS_i : modal state of individual (type I)
MS_c : cultural modal state (type II)
MS_m: modal states of the manifold (modal currents)

3.1 Three Types of MMF

The kind of quantities seen as fundamental in Type I and Type II can be regarded as derivative from modal currents. Individuals and their behavior are treated as being defined by a series of modal states of the individual (MS_i). Type II quantities may be reached by defining cultural modal states (MS_c).

Along with the progress, certain MS_c may grow or fade, new ones are introduced and so on. When various movements in societies are described in terms of the MMF, the corresponding MS_c are seen as derivative from the underlying modal states of the social manifold (MS_m), which are related to individual and collective components, and are not reducible to either of them.

The directional and dynamic nature of the MS_m through time will be referred within the manifold as modal currents. MS_i are derivative from the underlying modal states of the manifold (MS_m).

3.2 Features of Individual under MS_i

The following list are specific features of MS_i [1]:

1. Each MS_i has a cognitive function. It splits up or interprets the individual's stream of experience.

2. Each MS_i is associated with a set of images or symbols, by which that MS_i may be referenced or evoked.

3. Each MS_i corresponds to

   - specific moods, motivations, feelings and emotions.
   - a particular decision structure. Within it the individual will respond in certain ways to certain events, will subjectively find certain goals attractive and others unattractive.
   - a particular subjective sense of self and a particular way in which the individual perceives of his or her relationship to other individuals and other aspects of the environment.
   - certain physiological correlates, such as posture, muscle tension, blood pressure, etc..

4. MS_i differs in terms of possible transitions to other MS_i's. They may be more or less well connected to other states. They may also allow for more or less innovation in the creation of new states.

3.3 Bimodal and Multimodal Models

The framework of MMF offers a system schema involving structure and freedom. Structure exists at the level of the 'modal state' or conceptual framework and that of movement between frameworks. This is a substantial improvement on simple determinism.

3.4 Operation within the Flow

Actually, we have two levels of operation with the modal currents. One level consists of action (or interpreting the actions of others) within a given MS_i, i.e., a temporarily fixed framework of relationship. At the second level, we will operate with MS_i's themselves and move between possible frameworks of relationship.

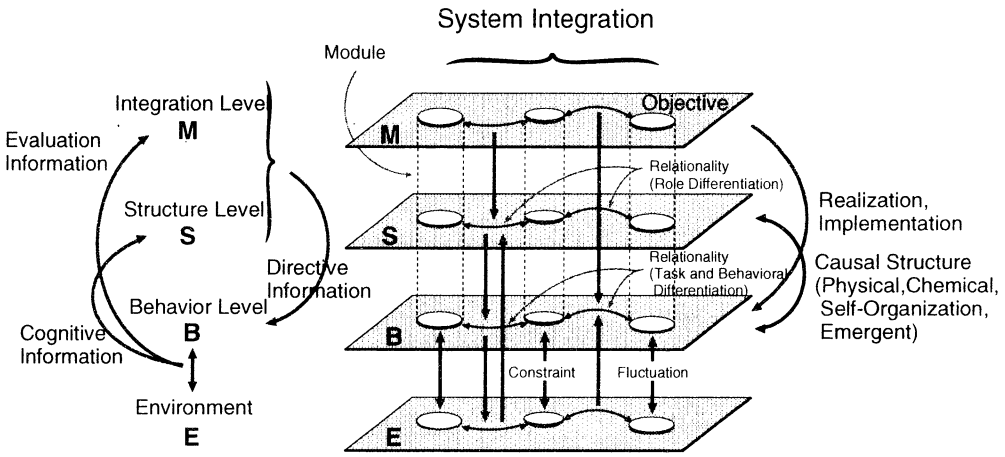


Figure 1: Bio-informatic modular system architecture

Table 1: System Concept: "Symbiotic Multi-body System of Modules"

Three Aspects of Modules	
Integration (**M**)	Value, Objective, Function·Role, Collaboration·Conflict
Structure (**S**)	Constraint·Boundary, Condition, Allowance of Behavioral Variety (Freedom)
Behavior (**B**)	Event Structure along with the Arrow of Time

Role of Information (Commitment)	
Cognition : Interaction with Environment	→ **S**
Direction : **S** , **M**	→ **B**
Evaluation: **B**, Interaction with Environment	→ **M**

4 Relation to Bio-informatic Multi-body Architecture

We have already introduced triple-layered multi-body architecture of life systems [4]. It consists of Integration Layer **M**, Structural Layer **S** and Behavioral Layer **B**. As shown in Table 1, we grasp on layer M the values, objectives and functions or roles of a module, and the collaboration or conflict among the modules can be analyzed. On layer S, constraint, boundary condition and allowance of behavioral variety are grasped. Also, the behavioral layer B represents the event occurrence structure along with the elapse of time.

Concerning this, T. Yoshida [5] introduced these basic "role of information" in his study on information theory of self-organizing systems. Namely, as shown in Table 1, Cognition, Direction and Evaluation are the main roles, and they respectively are interpreted as the function of linking interactions with environment with S, linking A and M with B, and linking B and interactions with environment with M.

Finally, we come to the system structure (system organization) which can be regarded as a "symbiotic multi-body system of modules" as shown in Fig. 1, where the notion of symbiosis introduced by K. Kurokawa [6] is examined by a research committee in Keidanren, which is summarized as the "Definition of Symbiosis": Generation and development of more than one entities with collaboration yielding an order formation in higher, wider and pluralistic values and dimensions [6].

The above mentioned anthropological framework, MMF, will provide us with various relationship among modules to form MS_i, MS_c and MS_m. For instance, decision structures and motivations will be related to a coordination structure on the Integration level (M). Physiological correlates, on the other hand, will be linked to coordinations on Structure level (S) or Behavioral level (B).

5 The Notion of Modal design

We now would like to introduce a novel notion of design, by confining the progress of design by setting up modal states and also modal currents among them. We will employ various modal logics [7] as the ba-

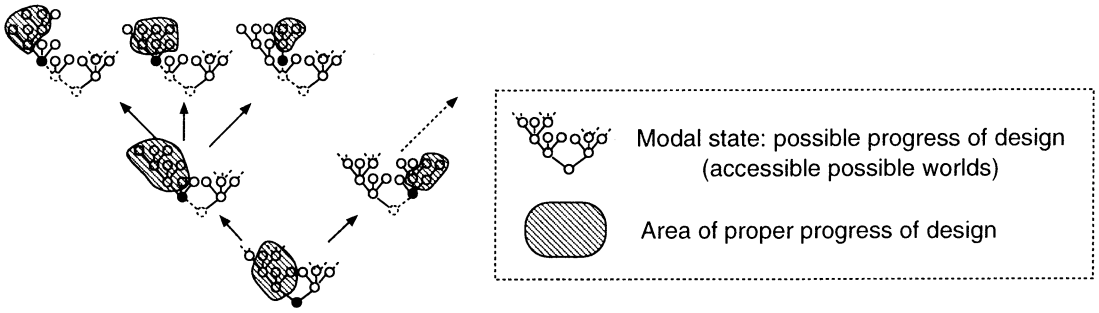


Figure 2: Conceptual description of the notion of Modal Design

sic frameworks for formalizing and representing the MMF.

Fig. 2 shows the schematic representation of the process of modal design, where admissible area of modal states are confined depending on the context and the progress of design.

It should be noted here that each modal state (together with other modal states) constitute the "context" of design, but also, the context in turn prescribes each modal state. This "reciprocal relation" is very important in ethnomodological approaches [8] where this reciprocality is referred to as reflexivity and indexicality.

In modal logics, each "possible world (state)" is associated with "accessibility relations" towards and from other possible worlds. We will regard each possible world as a modal state of an individual (even though, they are different in conceptual levels which may leads us to categorical mistakes). Also, the network of these accessibility relations corresponds to the manifold structure of these modal states, that is, the modal currents.

Also, the modal logical frameworks have tight connection with the Triple-layered modular architecture. Actually, the coordination on level M (integration) is tightly related to "deontic modalities", and the coordination on level B (behavior) will be naturally related to "temporal modalities".

The key issue in the modal design is the "logical consistency" of these constituents. This is also reduced to a kind of reciprocal relations, that is, the setting of each possible world such as the properties holding there and the accessibility relations from and towards it should be consistent with the setting of the neighboring possible worlds.

Idealizing the situation, this reciprocal consistency can be reduced to the above introduced notion of autoprescriptive set.

6 Conclusion

In this paper, we have introduced various novel notions such as Modal Design and Autoprescriptive Sets. These notions are utilized to extend our conceptualization of life systems as multi-body modularized symbiotic architecture. Modal Design is used to design these complex architectures, and the ways of solving autoprescriptive set equations can be utilized for progressing actual processes of modal design.

References

[1] G. Samuel, *Mind, body and culture*, Cambridge University Press, 1990

[2] D. McDermott and J. Doyle, "Non-monotonic logic I", *Artificial Intelligence*, **13**, pp.41-72, 1980

[3] R. Reiter, "A logic for default reasoning", *Artificial Intelligence*, **13**, pp.81-132, 1980

[4] O. Katai, "System Symbiosis with Humans, Artifacts and Environments via Coordinated Interactions and Communications", *The Journal of Three Dimensional Images*, **13**, 3, pp.101-109, 1999

[5] T. Yoshida, *Information Sciences on Self-Organizing Systems*, Shinyo-sha, 1990(in Japanese)

[6] K. Kurokawa, *The Philosophy of Symbiosis*, Academy Edition, London, 1994

[7] G. E. Hughes and M. J. Cresswell, *An Introduction to Modal Logic*, Mathuen and Co.Ltd, 1968

[8] K. Leiter, *A Primer on Ethnomethodology*, 1980

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15–17, January, 2001,

Reinforcement Learning based on Resonance between Agent and Field

Hideki Yamagishi*, Hiroshi Kawakami*, Tadashi Horiuchi** and Osamu Katai*

*Dept. of Systems Science, Graduate School of Informatics, Kyoto University
{yama/kawakami/katai}@sys.i.kyoto-u.ac.jp

**Institute of Scientific and Industrial Research, Osaka University
horiuchi@ar.sanken.osaka-u.ac.jp

Abstract

Recently, reinforcement learning methods have been successfully applied to various problems where latent rules cannot be observed nor acquired manually. Q-learning is one of the famous methods for reinforcement learning. One of the simplest ways to estimate Q-values is to look up a Q-table, but it cannot deal with continuous-valued inputs and outputs. We have already proposed a framework of reinforcement learning with Condition Reduced Fuzzy Rules (CRFRs) where Q-values are interpolated by fuzzy inference. In this paper, we apply Q-learning with CRFRs to a boat racing problem. Through the analysis of the experimental results, we show that the boat controller performs well without dominant fuzzy rules which have extremely high Q-value for selecting actions. We then will examine this curious result with introducing the notion of "synchronic firing structure" and that of "resonance between agents and field".

1 Introduction

Recently, reinforcement learning methods have been successfully applied to various kinds of problems [1]. In this learning framework, an agent tries to maximize the total amount of rewards which it receives from the environment. Q-learning [2] is one of the widely-used reinforcement learning methods, which employs Q-functions for evaluating condition/action pairs. One of the simplest ways to implement a Q-function is to look up a Q-table. Assume an environment that has n conditions for taking m actions. Each cell of the $n+m$ dimensional Q-table holds a value of one of the conditions/actions pairs (Q-value). The values are revised through the learning process. Q-table is simple, but its size will explode when it is applied to practical problems in which continuous-valued conditions and actions are involved. In addition, only a single cell of a Q-table is revised at each time of learning, and hence we cannot make good use of the continuity of condition/action values.

We have already proposed a Fuzzy Interpolation-Based Q-Learning which can deal with continuous inputs and outputs [3] and its improved version: Q-learning with Condition Reduced Fuzzy Rules (CRFRs) [4]. We applied Q-learning with CRFRs to a boat controlling problem and showed there are no dominant rules which have an extremely high Q-value. Through analyzing Q-values before/after learning, we showed that only valid rules are separated from a large amount of fuzzy rules and that those sets of rules contributing to the action selection can be extracted [5] which can control the boat quit well without dominant rules.

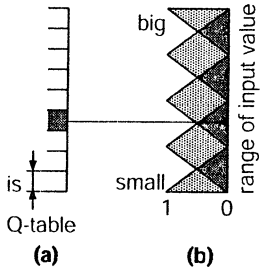


Figure 1: Fuzzy interpolation

In this paper, we will introduce the notion of "a synchronic firing structure" and the notion of "a resonance between a rule set for agent and that for field", and we will show that each fuzzy rule has the characteristics of either of the rule sets. We also discuss the way to inspect how the knowledge of field and that of agent resonate with each other.

2 Resonance between Agent and Field

2.1 Q-learning with Condition Reduced Fuzzy Rules

Fuzzy inference has been successfully used to interpolate values of Q-table [3]. This method estimates Q-values for conditions/action pairs by fuzzy inference. The framework of learning process is almost the same as the standard Q-learning algorithm [2] except for the ways of (1) estimating Q-values by using Takagi-Sugeno Model[6], of (2) selecting an action, and of (3) revising the Q-value of each fuzzy rule.

We have proposed fuzzy inference with Condition Reduced Fuzzy Rules (CRFRs) and incorporate it to Q-learning. The antecedent part of each CRFR consists of all the possible actions and the selected conditions and the consequent part shows the Q-value of the rule. Table 1 shows the parameters for generating the initial fuzzy rules. Let us assume that each condition and action involves fuzzy sets and that their numbers are uniformly set to be Nf. Fig-

Table 1: Parameters for generating initial fuzzy rules

Nc:	the total number of conditions
Nr:	the number of conditions which are included in each CRFR
Na:	the total number of actions
Nf:	the number of fuzzy sets for each condition and action
Ng:	the number of fuzzy sets having nonzero grades for an arbitrary input

Table 2: Framework of Q-learning with CRFRs

1. Initialize C_i (value of Q_i)
2. Repeat forever
   (a) Repeat T times
      i. Assume (select) a set of actions by random.
      ii. Calculate the grade of each rule (ω_i) under the current conditions and assumed actions. Matching grade ω_i of each fuzzy rule R_i is given by the algebraic product of the grades of its antecedents.
      iii. Estimate the Q-value for the current conditions/actions pair, i.e. $Q = (\sum_i \omega_i C_i)/(\sum_i \omega_i)$.
   (b) (after the second cycle)
      i. Calculate ΔQ for the last conditions/actions pair by the standard way of Q-learning, i.e., $\Delta Q = \alpha\{r_t + \gamma \max_b Q(x_{t+1}, b) - Q(x_t, a_t)\}$.
      ii. Distribute ΔQ to CRFRs so as to ΔC_i be proportional to ω_i, i.e., $\Delta C_i = (\sum_j \omega_j \omega_i \Delta Q)/(\sum_j \omega_j^2)$.
   (c) Select one of the actions by roulette wheel selection and execute it.
   (d) Observe the next state and the reinforcement signal.

ure 1(b) shows the case Nf = 5, Ng = 2 and the way how fuzzy membership function interpolates Q-tables shown in Figure 1(a). In case of Nc + Na = 3 and Nf = 5, the normal form of fuzzy rules will be represented as the following form:

$$\text{if } (X_1 \text{ is } P_{1y_1}) \wedge (X_2 \text{ is } P_{2y_2}) \wedge (X_3 \text{ is } P_{3y_3}) \text{ then} (Q_z = C_z),$$

where P_{xy}, Q_z and C_z denote fuzzy set, Q-variable and its value, respectively. The combination of y_1, y_2 and y_3 such that $1 \leq y_1, y_2, y_3 \leq 5$ yields 125 (=53) fuzzy rules. On the other hand, in case of Nr+Na = 2, the CRFRs will be given as

$$\begin{aligned} &\text{if } (X_1 \text{ is } P_{1y_1}) \wedge (X_2 \text{ is } P_{2y_2}) \quad \text{then} \quad (Q_l = C_l), \\ &\text{if } (X_2 \text{ is } P_{2y_2}) \wedge (X_3 \text{ is } P_{3y_3}) \quad \text{then} \quad (Q_m = C_m), \\ &\text{if } (X_1 \text{ is } P_{1y_1}) \wedge (X_3 \text{ is } P_{3y_3}) \quad \text{then} \quad (Q_n = C_n), \end{aligned}$$

where the total number of CRFRs is 75 (= 3 × 52). The framework of Q-learning with CRFRs is shown in Table 2.

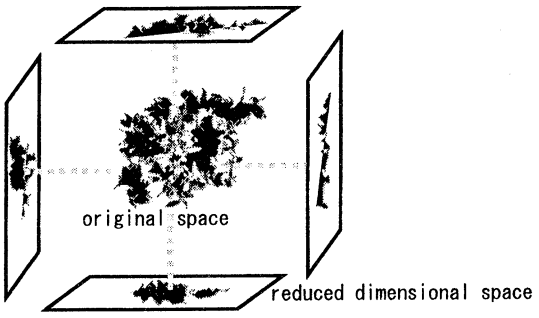


Figure 2: Conceptual image of synchronic firing structure

2.2 Separation of Behavior Modules

In the standard Q-learning, the controller learns Q-values in a Nc dimensional space, using all variables in the antecedent part of fuzzy rules. In Q-learning with CRFRs, the controller learns Q-values in ${}_{Nc}C_{Nr}$ reduced dimensional spaces (Nr dimensional conditions + Na dimensional actions) [5]. Q-values in a Nc dimensional space are estimated by fuzzy inference using backprojection of ${}_{Nc}C_{Nr}$ reduced dimensional spaces. Figure 2 illustrates a conceptual image of learning Q-values in four "reduced dimensional spaces". In Q-learning with CRFRs, the learning of Q-values proceeds on four planes rather than in the original space. As fuzzy rules synchronically fire on the four planes at each step, the actions for high Q-values are selected stochastically. The learning process makes some regions on the four planes emerge where fuzzy rules synchronically fire.

2.3 Coupling of Agent and Field

Generally, it has been assumed that the learning proceeds as "internal" process of an agent, and the environment is only the object to be observed. On the other hand, from the view point of Ecological Psychology [7], the relationship between agent and environment forms "the structural coupling" in natural way and invariants of the relation play an important role of acquiring the so-called "affordance". Introducing this notion of Ecological Psychology, learning is done in the process of coupling (meet) between agent and environment. The field of the coupling means not only the physical field (e.g. electric field, magnetic field, etc.) but also the ecological field such as living space. From this point of view, the coupling between "information for agent" which agent can sense and "information for field" is essential for reinforcement learning.

In the previous subsection, we discussed the learning proceeds on the ${}_{Nc}C_{Nr}$ reduced dimensional spaces in Q-learning with CRFRs. Here, we project the original high dimensional space to only two kinds of spaces; one is the space of information for agent and the other is the space of information for field. In this case, learning is regarded as forming a synchronically firing structure of "fuzzy rules for agent" and "fuzzy rules for field". In other words, learning proceeds to form a kind of the resonance between "knowledge of agent" and "knowledge of field".

3 Experiment

3.1 Experimental Environment

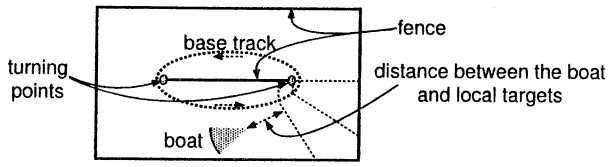


Figure 3: Experimental problem (boat racing track)

Figure 3 illustrates an experimental problem. Learning methods in this case will yield rules for controlling boats to go around the racing track. The state variables (condition for selecting an action) of this system are the current location (x, y), the velocity (v_x, v_y), the direction (r) and the angular velocity (w) of the boat, and the action is the combination of "steering wheel (hdl)" operation and "acceleration lever (acc)" operation. In this case, $Nc = 6$ and $Na = 2$. States are calculated so as to make the time-constant of the boat large, which makes the controlling of the boat quite difficult.

Fuzzy membership functions are set as shown in Figure 1 (b). Rewards in this case are set to be inversely proportional to the distance between the boat and the nearby local target as shown in Figure 3, and each time a boat collides with a fence, it will be penalized. Penalties are given as rewards with a certain negative value.

3.2 Experimental results

3.2.1 Synchronic Firing Structure

Figure 4 shows the learning performance of Q-learning with CRFRs. Figure 4 reveals that the result with CRFRs is equivalent to or even better than those with the most detailed rules ($Nr = 6$). In the case where $Nr = 3$, there are 20 (${}_6C_3$) reduced dimensional spaces and Figure 5 shows one of them. From this figure, we can see that there are no dominant rules but that rules having high Q-values are packed.

On the other hand, Figure 6 shows the distribution of firing rules shaded by the value of matching grade × Q-value at a certain step of driving the boat. From this figure, we can see that several rules synchronically fire on one plane at each step. We confirmed that at each step of action selection, a few rules synchronically fire on all planes. This indicates that the area where rules synchronically fire emerge on each plane through the learning process.

3.2.2 Separation of Rules for Agent and Field

In the previous subsection, Nc dimensional Q-value space is projected to twenty planes, which are simply separated according to the kind of variables of antecedent part of fuzzy rules. In this section, we introduce the notion of resonance between agents and field. That is to say, Nc dimensional Q-value space is projected to only two planes which govern the fuzzy rules for agents and that for field.

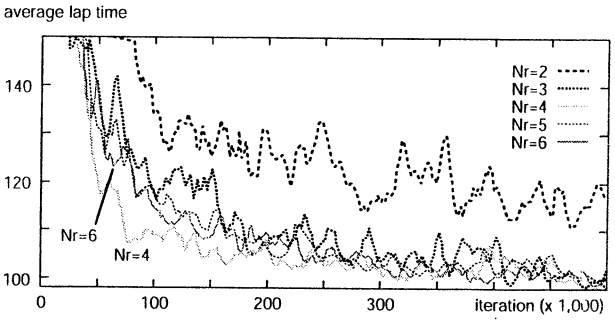


Figure 4: Learning performance

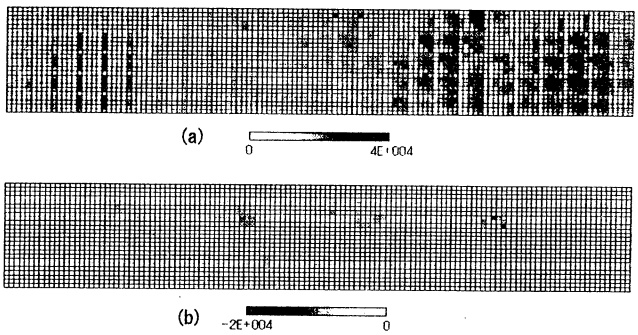


Figure 5: Distribution of Q-values

Hence, we discuss a method to separate fuzzy rules into two planes instead of twenty planes.

First of all, the agent of boat A learns the fuzzy rules R_A. Then, the agent of boat B which has a different time-constant, begins learning by using fuzzy rules R_A as the initial fuzzy rules. After learning of agent B, the fuzzy rules can be classified into the following four groups. Figure 7 shows an example of experimental results. First group (Region I) means the set of fuzzy rules each of which gets rewards through both learning processes for boats A and B. Second group (Region II) means a set of fuzzy rules that are penalized by both learning processes. These two groups are independent of agent, and hence we call the rules in these groups "rules for field". On the other hand, the rules in Regions III and IV are for the agent because their utility are dependent on the agent. For example, the rules in Region III are beneficial for controlling boat A, so they get rewards but they are penalized through controlling boat B.

4 Discussion

In the previous chapter, through analyzing the Q-value of each rule, we have shown that the boats can be controlled by a synchronic firing structure of fuzzy rules. However, there are so many firing rules that it is difficult to find out overall structure of the resonance. A Q-value for taking an action under a certain situation is calculated by Q-values of many fuzzy rules. Thus, from a broad viewpoint, we will discuss the way for categorizing parameters into two types by using Q-values for taking actions. The one category governs agents and the other governs the field. Usually,

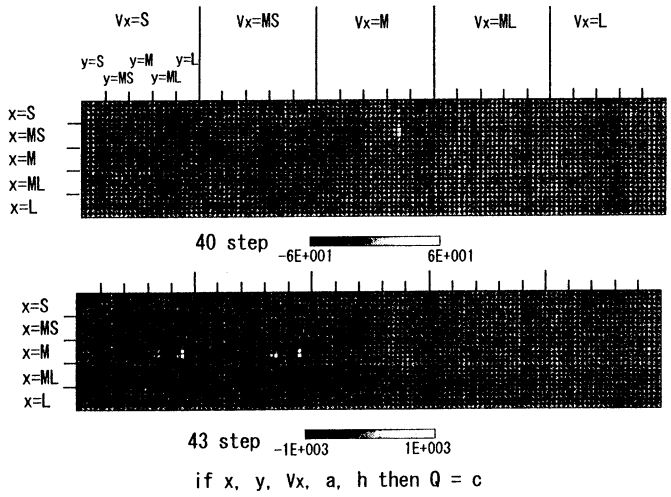


Figure 6: Distribution of firing rules

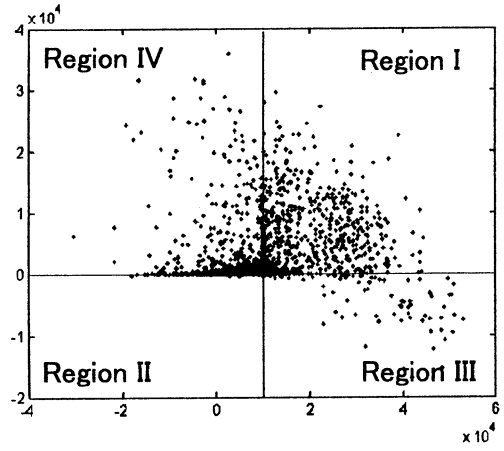


Figure 7: Rules of agent and field

the probability of taking a set of action (act) is calculated by

$$P(act|x) = \frac{exp(Q(x, act)/T)}{\sum_b exp(Q(x, b)/T)}.$$

In the case where a set of state parameters x can be classified into two types (say, Agent and Field), the left hand side of the above equation can be transformed by using the Bayse theorem $P(\alpha|\beta) = P(\alpha, \beta)/P(\beta)$ into

$$P(act \mid \text{Field}, \text{Agent}) = \frac{P(act, \text{Agent} \mid \text{Field})}{P(\text{Agent} \mid \text{Field})}.$$

The denominator of this equation means that along with the learning process, the probability of agent's state is getting deterministic under a certain state of the field. The numerator of this equation can be interpreted as the same way as the denominator.

5 Conclusion

The "knowledge of agent" and "knowledge of field" correspond to the *modal state of individual* (MSi) and *cultural modal state* (MSc) in the *multimodal framework* (MMF [8]) which is proposed in the researches on the social and cultural anthropology. Originally, modal states are introduced in order to escape from mind-body dichotomy. Modal states can be thought of as providing a description of the semi-permanent currents or vortices. MSi (the area of so-called "individual" and "body") corresponds to a pattern with particular 'density' or 'intensity' separated out from the currents or vortices. The vortices denote 'connectedness' among human beings or between human beings and ecological system. MSc (the area of so-called "society" and "culture") cannot be explained by monoismistic framework. It is a polymerization of some modal states which synchronically compete and diachronically change over. Here, "knowledge of field" and "knowledge of agent" correspond to MSc and MSi respectively.

In this paper, we applied Q-learning with CRFRs to a boat racing problem and explained the reason why the agent can control the boat well without dominant rules for selecting actions. Through analyzing Q-values of each rule, we showed a synchronic firing structure of fuzzy rules and categorized rules into "rules for agent" and "rules for field". Furthermore, based on the discussion of probabilities of selecting actions, we introduced a novel point of view of "resonance between agent and field".

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An introduction*, The MIT Press, 1998.

[2] C. Watkins and P. Dayan, "Technical Note: Q-learning", *Machine Learning*, **8**-3/4, pp.279-292, 1992.

[3] T. Horiuchi, A. Fujino, O. Katai and T. Sawaragi, "Fuzzy Interpolation-Based Q-Learning with Continuous Inputs and Outputs "(in Japanese), *Trans. of the Society of Instrument and Control Engineers*, **35**-2, pp. 217-279, 1999.

[4] H. Kawakami, T. Konishi and O. Katai, "A Reinforcement Learining with Condition Reduced Fuzzy Rules", *Proc. of the 2nd Asia-Pacific Conference on Simulated Evolution and Learning*, LNAI 1585, Springer, pp. 198-205, 1998.

[5] H. Yamagishi, H. Kawakami, T. Horiuchi and O. Katai, "Separation of Behavior Modules by Reinforcement Learning with Condition Reduced Fuzzy Rules," *Proc. of the 4th Japan-Australia Joint Workshop*, pp. 114-121, 2000.

[6] M. Sugeno, *Fuzzy Controls* (in Japanese), Nikkan Kogyo Shinbun, 1988.

[7] John Flach, Peter Hancock, Jeff Card, Kim Vicente, *Global Perspectives on the Ecology of Human-Machine Systems*, vol. 1, Lawrence Erlbaum Associates, 1995.

[8] Geoffrey Samuel, *Mind, body and culture*, Cambridge University Press, 1990.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Affordance based Human Motion Synthesizing System

H. Ishii, N. Ichiguchi, D. Komaki, H. Shimoda and H. Yoshikawa
Graduate School of Energy Science
Kyoto University
Uji-shi, Kyoto, 611-0011, Japan

Abstract

A human motion synthesizing system has been developed for generating various kinds of human motions flexibly as 3 dimensional computer graphics in virtual environment. This system is designed based on the idea derived from the concept of affordance. The idea is that the entire algorithms and the information necessary for synthesizing a human motion should be composed in the object database which is an archive for the virtual object's information. This design methodology makes it possible to make a new algorithm for synthesizing a human motion available without reconstructing the human motion synthesizing system. In this paper, how to apply the concept of affordance to design the human motion synthesizing system and the overall configuration of the developed system are described.

1 Introduction

The goal of this study is developing a new training system which is realized by combining the Virtual Reality (VR) technology and the Artificial Intelligence. The authors call the training system "Virtual Collaborator" and have made studies such as [2, 4]. The Virtual Collaborator provides an artificial instructor who has a human-shaped body and can listen, talk, think, behave and collaborate with real humans. The artificial instructor helps a trainee learn complicated tasks by instructing and demonstrating them in a virtual space. In our previous study[2], a prototype Virtual Collaborator has been developed in which the artificial instructor can behave just like a plant operator in the control room of nuclear power plant. But some problems have arisen at developing the advanced Virtual Collaborator with which the trainee can collaborate with the artificial instructor through bi-directional communication[4].

Firstly, it is very difficult to synthesize various kinds of the artificial instructor's motions as 3 dimensional computer graphics in real time. A human has a lot of joints such as neck, shoulder, elbow, wrist, waist etc. and each joint has from one to three degrees of freedom (DOF). So a human has a large number of posture variables. To synthesize the human motion, all of the joint's angles must be specified. Numerous algorithms for synthesizing human motions can be found in literature, but all of them are limited to use for synthesizing a particular motion. Therefore, to make it possible to synthesize a new kind of human motion, a new algorithm is needed to develop and to make it available. On the other hand, it is impossible to prepare all the algorithms necessary for synthesizing the artificial instructor's motion by predicting which kinds of motion will be necessary in the future. So it is inevitable to develop a new algorithm whenever a new kind of human motion needs to be synthesized.

Secondly, it is very difficult to execute the training simulation in real time, because the vast computation load is required. To execute the training simulation, it is necessary to synthesize the body motion of the artificial instructor, generate a virtual space as 3 dimensional images and execute the human model simulator as the artificial instructor's brain.

In this study, to solve these problems, the authors developed an Affordance based Human Motion Synthesizing System (AHMSS) which is designed based on the idea derived from the concept of affordance, introduced by psychologist James Gibson[5]. In which follows, described are how to apply the concept of affordance to design a new human motion synthesizing system and how to configure the whole-developed system.

2 The concept of affordance and its application for the system development

The conventional method of developing a system using computer animation of virtual humans has been a way like this; first what kinds of the virtual human's

motion should be synthesized for realizing the system is decided, and then the algorithms and the data for synthesizing those kinds of virtual human's motion are constructed into the system. This is the way that the surrounding environment the virtual human is located is decided first and then the knowledge about the environment is created and put into the virtual human as the model of environment. Of course, even by this, the virtual human can behave in accordance with the knowledge about the environment. But it is very difficult to prepare all the knowledge in advance about the environment the virtual human could be located in the future.

As one solution to this problem, there is the concept of affordance. The affordance was introduced by psychologist James Gibson and he defined the affordance as "a specific combination of the properties of substance and its surfaces taken with reference to an animal." According to this concept, an action of a human is triggered by the environment itself where the human exists unlike the afore-mentioned way of interpretation that the human would behave in accordance with the model of the environment the human already possesses in advance.

When this way of thinking would apply to the development of the human motion synthesizing system, the algorithms and the data for synthesizing the virtual human's motion should be composed not in the virtual human's brain but in the virtual objects located in the virtual environment. And the algorithms and the data should be transferred from the virtual object to the synthesizing system at the time when they become necessary.

For example, a floor affords "walk-on-ability" to the virtual human if the floor is large enough and smooth enough. In this case, the algorithms and the data for synthesizing the walking motion should be composed not in the synthesizing system but in the database which describes the information about the floor. In other words, the necessary information for synthesizing the virtual human's motion should not be composed in the synthesizing system but in the database which describes the information about the virtual objects such as the 3 dimensional shape, texture and so on.

As mentioned above, by composing all the information necessary for synthesizing the virtual human's motion into the virtual object, there are some advantages as follows:

(1) Because it becomes possible that all the information related to one virtual object could be put together being separated from the other virtual object, it is easy to add a virtual object into the virtual environment.

(2) By editing the database for the virtual objects, it is possible to make an algorithm for synthesizing the virtual human's motion available without reconstructing the system.

Based on the discussions mentioned above, the authors make it the first policy of the system design that the algorithms and the data necessary for synthesizing the virtual human's motion are composed in the database not for the virtual human but for the virtual objects.

3 Requirements

In this chapter, the requirements the AHMSS should satisfy as a system for synthesizing the virtual human's motion are described. In this study, in consideration of the design principle derived from the concept of affordance mentioned in chapter 2 and the situation the AHMSS is used as a component of the advanced Virtual Collaborator, the authors designed the AHMSS to satisfy the following 4 requirements:

(1) Both of the virtual human's motion and the virtual object's movement can be synthesized at the same time.

To develop the advanced Virtual Collaborator as a personalized interface, it is necessary for the artificial instructor not only to communicate with real humans by gestures but also to manipulate virtual objects with his both hands. So it is necessary to synthesize not only the virtual human's motion but also the virtual object's movement.

(2) It is possible to make a new algorithm available for synthesizing the virtual human's motion without reconstructing the system.

In the AHMSS, as mentioned in chapter 2, the information necessary for synthesizing the virtual human's motion is composed in the object database for the virtual object which is the target of the virtual human's motion. This system structure makes it possible to add a new algorithm to the AHMSS without reconstruction. But there are a lot of cases where the same algorithm or the same database is necessary for synthesizing the different motions. Therefore, in this study, an algorithm database which is an archive of the algorithms is introduced into the AHMSS and only

the name of the algorithm and the database for synthesizing the virtual human's motion is composed in the object database.

(3) The users of the AHMSS indicate the kind of the virtual human's action via a terminal.

In the advanced Virtual Collaborator, the AHMSS synthesizes the virtual human's motion in accordance with the indication of the Human Model. But the Human Model has not been constructed yet, so the AHMSS is designed that the indication to the virtual human is given from the user via a terminal.

(4) The AHMSS can synthesize the virtual human's motion and the virtual object's movement in real time.

To realize the advanced Virtual Collaborator as a personalized interface, it is necessary to update the virtual environment fast enough so that the user does not feel incongruous by looking the artificial instructor's motion. In this study, the authors designed the AHMSS to realize the parallel and distributed processing by separating the computational load into 3 processes of computation: the virtual human's motion, the virtual object's movement and the generation of the 3 dimensional images of the virtual environment. In the AHMSS, the 3 processes are executed on the 3 different workstations which are connected via network.

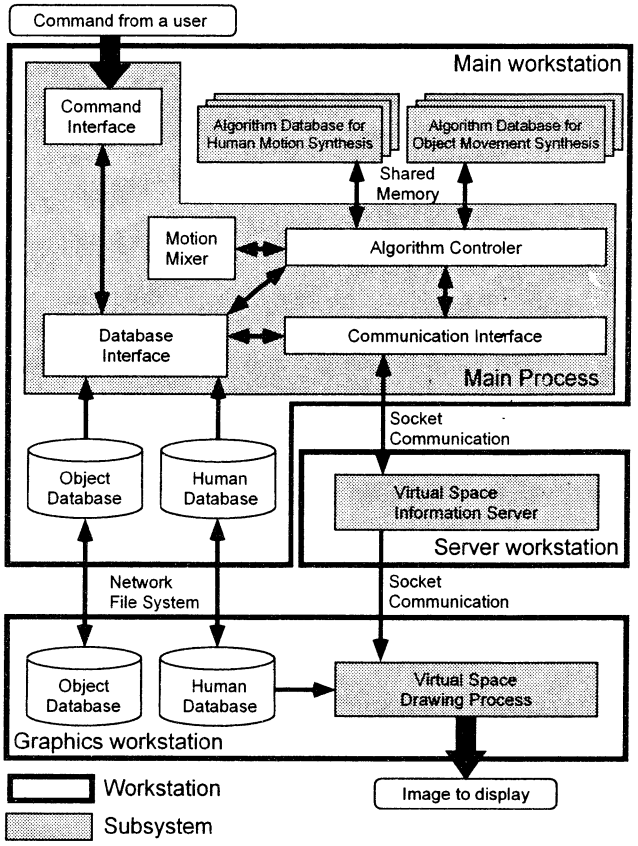


Figure 1: Configuration of the AHMSS.

4 System configuration

In this chapter, the configuration of the AHMSS is described. As shown in Figure 1, the AHMSS consists of 3 subsystems: Main Process, Virtual Space Information Server, Virtual Space Drawing Process and 4 databases: Object Database, Human Database, Algorithm Database for Human Motion Synthesis and Algorithm Database for Object Movement Synthesis. These subsystems are executed on three kinds of workstations: Server workstation, Main workstation and Graphics workstation, which are connected via network. The details of the subsystems and the databases are explained below.

(1) Algorithm Database for Human Motion Synthesis and Object Movement Synthesis

The Algorithm Database for Human Motion Synthesis and Object Movement Synthesis are archives of the algorithms for synthesizing the virtual human's motion and the virtual object's movement respectively. These algorithms are developed as the programs which can be executed on a unix workstation independently of the other algorithms and subsystems. As the algorithms for synthesizing the virtual human's motion, the algorithm "grasp an object" and "maintain a posture of the arm" have been developed besides the algorithms explained as follows:

- Motion capture

  This algorithm synthesizes the virtual human's motion by using a sequence of human postures obtained by measuring the motion of a real human with 3 dimensional motion capture system.

- Walking synthesis

  This algorithm was originally developed by the authors[3] and can synthesize walking motion of arbitrary direction and distance.

– Spherical cubic interpolation (Key-framing)
This algorithm synthesizes the virtual human's motion by the way that the motion is recorded as a sequence of key-postures and the computer reconstructs the motion by interpolating intermediate postures from appropriate key-postures.

(2) Main Process

The Main Process consists of Command Interface, Motion Mixer, Database Interface, Algorithm Controller and Communication Interface. The Main Process accepts commands from the user via the Command Interface and selects appropriate algorithms from the Algorithm Database in accordance with the commands and starts the algorithms as external processes. Then the necessary information for synthesizing the virtual human's motion and the virtual object's movement are sent to the processes via shared memory and the calculation results are returned to the Main Process. The Motion Mixer mixes two kinds of the virtual human's motions in accordance with the prepared weighted average[1]. The Main Process sends the results to the Virtual Space Information Server.

(3) Virtual Space Information Server

The Virtual Space Information Server manages the information about virtual environment such as the location and posture of the virtual human and the location and orientation of the virtual objects. The Virtual Space Information Server sends these informations to the Main Process and the Virtual Space Drawing Process by their requests. And these informations are updated in accordance with the calculation results from the Main Process.

(4) Virtual Space Drawing Process

The Virtual Space Drawing Process generates 3 dimensional images of the virtual human and the virtual objects in real time in accordance with the information about the location and posture of the virtual human and the location and orientation of the virtual objects from the Virtual Space Information Server.

(5) Object Database

The Object Database stores the information about virtual objects located in the virtual environment. As shown in Figure 2, the Object Database includes various kinds of the information about virtual objects such as virtual object's name, 3 dimensional shape, the action name the virtual object affords, the algorithm name for synthesizing the virtual human's motion, the algorithm name for synthesizing the virtual object's movement and so on.

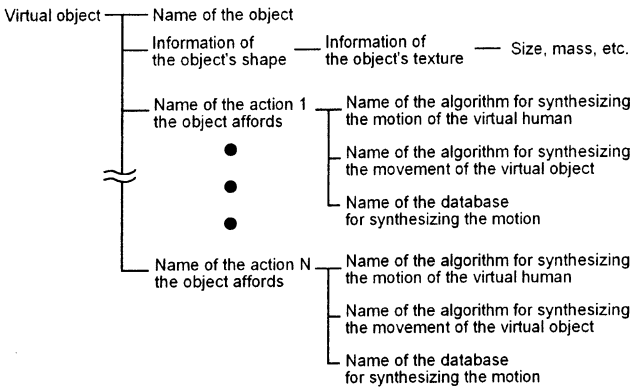


Figure 2: The Structure of the Object Database.

(6) Human Database

The Human Database stores the information about the virtual human located in the virtual environment, such as 3 dimensional shape of the virtual human's body, textures, the weight and the length of the body parts and so on.

The procedure for synthesizing the virtual human's motion in accordance with the indications from the user is shown in Figure 3 and summarized as follows:

Step1 The user allocates virtual objects and a virtual human into the virtual environment.

Step2 The user indicates the virtual object which is the target of the virtual human's action.

Step3 The system searches the Object Database for the indicated virtual object and shows a list of actions the indicated object affords.

Step4 The user selects an action from the list of actions and inputs the information necessary for synthesizing the virtual human's motion.

Step5 In the case of mixing two actions, repeat Step2, 3 and 4.

Step6 According to the indicated actions, appropriate algorithms for synthesizing virtual human's

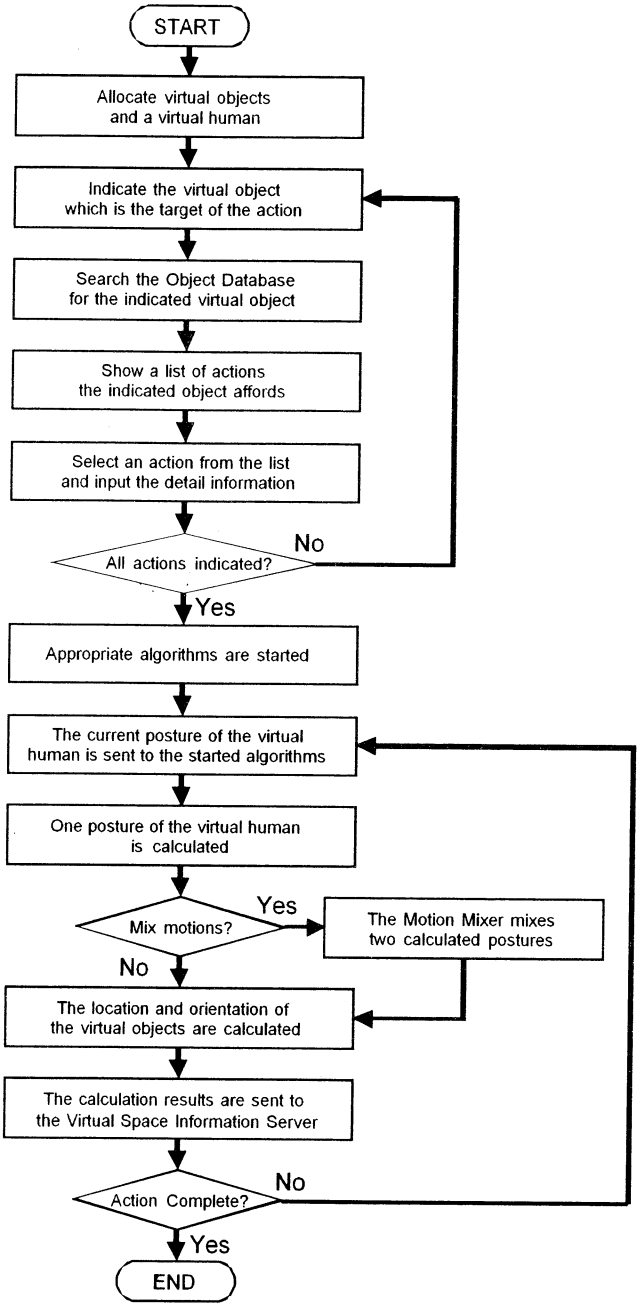


Figure 3: The Procedure for Synthesizing the Virtual Human's Motion.

motion and the virtual object's movement are started. In the case of mixing two actions, 2 algorithms for synthesizing the virtual human's motion and 2 algorithms for synthesizing the virtual object's movement are started.

Step7 The current posture of the virtual human is sent to the started algorithms for synthesizing the virtual human's motion.

Step8 The started algorithms for synthesizing the virtual human's motion calculate one posture of the virtual human.

Step9 In the case of mixing two actions, the results of Step8 are sent to the Motion Mixer and two postures are mixed according to the weighted average.

Step10 The posture of the virtual human calculated in Step8 or Step9 is sent to the started algorithms for synthesizing the virtual objects' movement.

Step11 The started algorithms for synthesizing the virtual objects' movement calculate the locations and orientations of the virtual objects.

Step12 The results of Step10 and Step11 are sent to the Virtual Space Information Server.

Step13 Repeat from Step7.

Figures 4 and 5 show the example motion synthesis of the virtual human who picks up a cup while walking and drinks water while walking respectively. In this study, the AHMSS was implemented on a Linux Workstation (Pentium III 700MHzx2) as the Main Workstation, a SGI Octane (R10000 250MHz) as the Graphics Workstation and a SGI O2 (R10000 250MHz) as the Server Workstation. As a result, it was confirmed that the developed system satisfies all the requirements described in chapter 3.

5 Concluding remark

In this study, an Affordance based Human Motion Synthesizing System (AHMSS) has been developed based on the idea derived from the concept of affordance which is one of the important concept in the field of cognitive science. The AHMSS was designed so that the algorithm and the necessary information for synthesizing the virtual human's motion are composed in the object database which is an archive for the virtual object's information. This design methodology

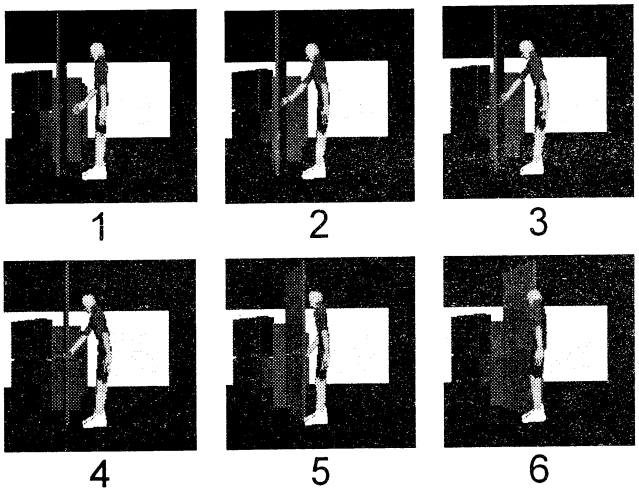


Figure 4: The example snapshots of the virtual human who opens a door.

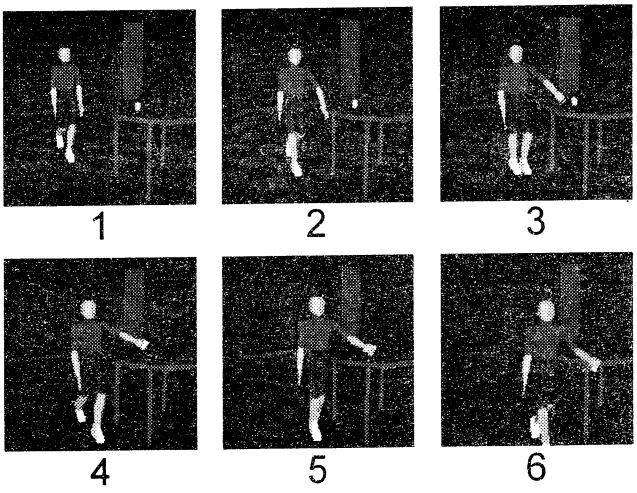


Figure 5: The example snapshots of the virtual human who picks up a cup while walking.

makes it possible to add a new kind of the algorithm for synthesizing the virtual human's motion without reconstructing the system.

For the future work, more algorithms for synthesizing the virtual human's motion and the virtual object's movement should be developed because it is necessary to synthesize more kinds of virtual human's motions to realize bi-directional communication between real and virtual humans with the advanced Virtual Collaborator. Moreover, the Graphical User Interface for editing the object database and allocating virtual objects into the virtual environment should be developed.

Acknowledgements

We gratefully acknowledge financial support from the Japan Society for the Promotion of Science under the research for the future program (JSPS-RFTF97I00102).

References

[1] Douglas,D. and Sundhanshu,S, "Fast Techniques for Mixing and Control of Motion Units for Human Animation", *Proceedings of Graphics'94*, pp. 229-242, 1994.

[2] H. Ishii, W. Wu, D. Li, H. Ando, H. Shimoda, T. Nakagawa and H. Yoshikawa, "A Basic Study of Virtual Collaborator - The First Prototype System Integration", *Proceedings of the 4th International Symposium on Artificial Life and Robotics*, Vol. 2, pp. 682-685, 1999.

[3] H. Shimoda, H. Ando, D. Yang and H. Yoshikawa, "A Computer-Aided Sensing and Design Methodology for the Simulation of Natural Human Body Motion and Facial Expression", *Proceedings of EDA'98*, CD-ROM, 1998.

[4] H. Yoshikawa, H. Shimoda, W. Wu, H. Ishii and K. Ito, "Development of Virtual Collaborator as an Innovative Interface Agent System between Human and Plant Systems: Its Framework, Present Status and Future Direction", *Proceedings of the 5th International Symposium on Artificial Life and Robotics*, Vol. 2, pp. 693-698, 2000.

[5] J. Gibson, *The Ecological Approach to Visual Perception*, Houghton Mifflin Company, 1979.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Interfacing and Co-Adaptation between Interaction Loops of Human- and Machine-Autonomies Mediated by Ecological Task Environment

Tetsuo Sawaragi and Yukio Horiguchi
Dept. of Precision Engineering, Graduate School of Engineering, Kyoto University
Yoshida Honmachi, Sakyo, Kyoto 606-8501, Japan.

Email. sawaragi@prec.kyoto-u.ac.jp and Yukio Horiguchi¡t60x0141@ip.media.kyoto-u.ac.jp¿

Abstract

This paper presents a new style of human-robot collaboration in a teleoperating environment based upon their mixed-initiative interactions, where both of them can have the initiative to control the system depending on the situation while keeping their independence. In order to provide with the naturalistic and intuitive relationship to the operator, our system allows the operator's variable actions including both the performatory and epistemic actions. The interactions performed by the collaboration are analyzed using Brunswik's Lens Model, and the roles of mediating environmental factors for the human-robot collaboration are discussed.

1 Introduction

Autonomous systems such as nervous systems of biological systems can be characterized by their self-organizing capabilities driven mainly by the internal "coherence" produced by their internal mutual relations, rather than be described by inputs from an external environment [6]. In the conventional cybernetics views, Environment has been regarded as sources of perturbations, and the main goal of the desired system has been thought as keeping a system's stability by protecting themselves by avoiding and isolating these externally provided perturbations. On the other hand, in living biological systems, such disturbances are essential and are used as sources of creativity; living systems proactively form significant relations with those through iterative and cyclic self-organizing processes and by absorbing them within them.

In order to realize such a process, living systems are endowed with the following characteristics: Plasticity of structure and continuous production of variable behaviors in reply to the change in environment. At the same time, structure also has to have stability of not being affected by such noisy changes in the environment. This stability and plasticity are indeed contradicting functions, but living biological systems are actually realizing both of them by letting their system components constitute organizational "resources" that either improve the overall order in the system by joining some coherent ensembles, or degrade that order by disengagement. As a result of this, partitioning and integration of various functions are realized. Our main goal of this work is to design such a unique process of living systems for maintaining internal coherence through continuous and dynamic productions and diminishing of relations within human-robot interactions.

For this purpose, we would like to stress the following two issues as distinguishing technical problems from the ones approached so far. One is "co-adaptation", in which we have to acknowledge that both the human autonomy and the machine autonomy like a robot should *co-evolve*, and both of them should be *co-adaptive* to each other. The conventional design principle for human-machine systems has been regarded as optimizing the fixed, separated task-allocation between the human-autonomy and the machine autonomy (i.e., automation). That is, the relations between human and machine have been often discussed as the relations between supervisor and supervisee, and/or between teaching-signal provider and obedient learner. In contrast to that, we would like to shift our perspective towards how to maintain the human commitment in the interactions with the target systems and how to stimulate the human participation within the loop of such interactions. The second issue is on the interface design for the former goal. Human actions or operations have been popularly analyzed by the conventional studies of human interface, but most of those works have assumed that the human user only takes such intentional and/or purposive actions, and discussions or analysis from the perspectives of more global *activities* permitting a variability of operator actions such as *epistemic actions* was not done.

In the followings, we at first describe the characteristic *socialization* process of the living system, and would like to extend this principle to the design of human-robot collaboration systems.

2 Biologically-Inspired Design

2.1 Co-Adaptation of Living Systems: Socialization Process

Due to the existence of the intentional others, any living systems cannot completely control the results that are brought about through the interactions with the external, while it gradually and continuously changes its own self, not fixated. This is a "socialization" process in sociology. That is, by interacting with the others

and reflecting on this experience, living systems reconstruct a definition of the reality, which causes them behave differently and changes the interaction patterns. This is iterated indefinitely, so the living system keeps changing and produce a variability. What is important here is not what will be acquired finally, but this cyclic process per se is important. By iterating this cycles, the living actor, which is equivalent to the perceiver and the observer, comes to be embedded within the real world, and what are acquired within the actor becomes *grounded* to the world. This is a general learning model of *observing* systems, rather than *observed* systems, and have a continuous self-production process enabled by evolutional development through its internally-inspired activities. Through such a bidirectional socialization process, living systems come to be engaged in perceiving and forming some a analogy and/or morphology to the others, which is sometimes called as a formation of social domain.

2.2 Epistemic Actions as a Social Act

Succeeding the above works, in this paper we present a novel architecture for a robot teleoperation system that is based upon the above mentioned socialization process of living systems. In a teleoperating environment, embedded autonomy in a remote system (i.e., a robot) will be indeed an effective approach to improve the overall system performance. In order to establish the naturalistic collaboration between a human operator and an autonomous robot, it is significant to consider how to let them share their recognition of the situation and their intentions. Especially their interaction and collaboration styles must be designed as coherent with the way of human *cognition*. Human cognition per se cannot be defined as *closed* within the individual's internal, but it must be shaped by the environment including the collaborating partner, the external world and the internal states of the self (i.e., distributed cognition). From this perspective, the modeling approach must deal not only with the isolated systems of the human operator and the autonomous robot, but also with the interactions made among them.

Concerning with the human cognition, the new aspect of action is recently focused. That is, action plays not only a performatory role but also an *epistemic* or *knowledge-granting* role. This proactive aspect of action is necessary to the flexible and skillful performance of a human in the complex world because it can be the efficient strategy to reduce his/her cognitive burden to infer. But in a conventional human-machine system design, an operator's action is extremely restricted in the control loop of the highly automated system for its stability, security, and so on. In order to provide with the naturalistic and intuitive relationship to the operator, the system should allow the operator's variable actions including his/her epistemic actions. Based on the above ideas, we propose a new style of human-robot collaboration in a teleoperating environment based upon their *mixed-initiative* interactions, where both of them can have the initiative to control the other depending on the situation while keeping their independence.

In this paper, based on the above ideas, we propose a new style of human-robot collaboration in a teleoperating environment based upon their *mixed-initiative interactions* [4], where both of them can have the initiative to control the system depending on the situation while keeping their independence.

3 Analysis of Robot Teleoperating Environment

3.1 Lens Model as Analysis Method

Brunswik's Lens Model [1] is a functional representation of human perception and judgment that can describe their causal relationship without separating his/her internal and external state. This property of the model is important for us to model and analyze a teleoperating environment as mentioned in the introductory section.

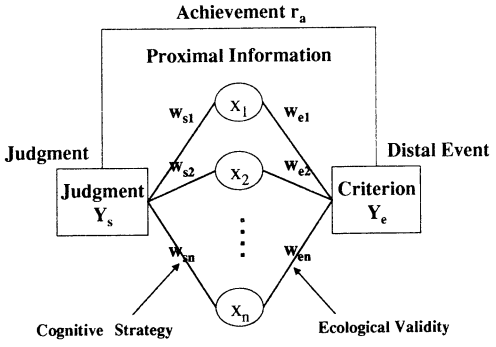


Figure 1: Brunswik's Lens Model.

Figure 1 shows the overview of this model. As shown in the center and right side of the figure, the task environment is described as consisting of three elements in the model: a) *proximal information* (or cues) that can be directly perceived and measured by a human; b) *distal event* that must be inferred based on proximal information because it cannot be directly perceived; and c) *ecological validity* that represents the constraints between the values of proximal information and distal event.

On the other hand, the left side of the figure describes the human *judgment* about the distal event. This judgment is made on the basis of *cognitive strategy* that expresses how he/she utilizes the proximal information $(X_1, X_2, \ldots, X_n)$ to infer the distal event.

The cognitive strategy is represented as connections between the proximal information and the judgment as well as the ecological validity between the proximal information and the distal event. Their weights (i.e., w_s and w_e) are usually regressed from an accumulated data set of cues, the performer's judgment, and the actual environmental state. If the judgment achieves to be successful, the weights of the cognitive strategy will have the similar values of the ecological validity. The correspondence between the judgment (Y_s) and the actual value of the distal event (or the criterion Y_e) is measured by the achievement parameter r_a. [1]

The Lens Model makes a distinction between proximal and distal information only in human perception (i.e., the proximal/distal structure of perception), so

[1] $r_a = 1$ means the perfect match between Y_s and Y_e

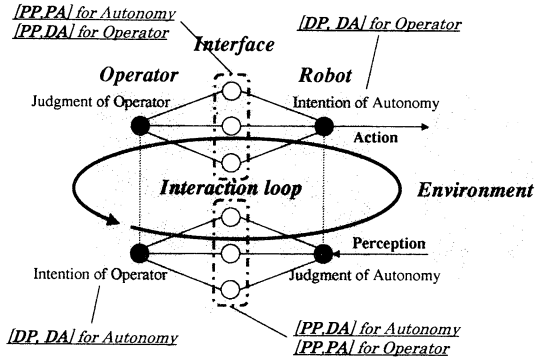


Figure 2: The analysis of a teleoperating environment.

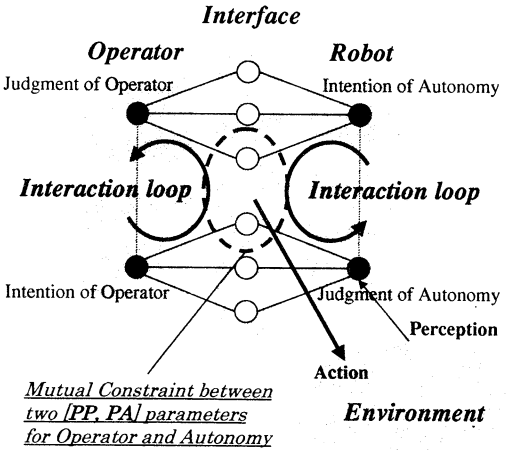


Figure 3: The collaboration style in our model.

Kirlik has proposed to add the proximal/distal structure of action into the Lens Model as his *Generalized Lens Model* [3]. In this model, variables in the task environment are considered to be proximal or distal from the perspective of action if the performer can directly manipulate their values or not.

With this extension of the model, variables in the task environment can be classified into the following four different types of information:

I [PP,PA]: a variable that is proximal for both perception and action (i.e., proximal perception and proximal action);

II [PP,DA]: a variable that is proximal for perception but distal for action (i.e., proximal perception and distal action);

III [DP,PA]: a variable that is distal for perception but proximal for action (i.e., distal perception and proximal action); and

IV [DP,DA]: a variable that is distal for both perception and action (i.e., distal perception and distal action).

After this framework, we will classify the variables concerning with the teleoperating task to analyze a robot teleoperating environment.

3.2 Analysis of Robot Teleoperating Environment

Figure 2 shows our analysis result of a teleoperating environment comprised of a human operator and an autonomous robot. In their intent sharing, the intention of the robot is a [DP,DA] variable for the operator, (the upper lens model of the figure) as well as the intention of the operator is [DP,DA] for the robot (the lower one of the figure). In order to judge the robot's intention, the operator utilizes the information's on the interface devices that are [PP,DA] for him/her and [PP,PA] for the robot as well as the robot does.

Through this interaction style, it is difficult to realize the naturalistic and intuitive collaboration between the operator and the robot due to the large time lag as well as to the operator's deliberative reasoning burden because of the difficulty to associate his/her action with the changes of his/her perception responding to it. Therefore it is necessary to design smaller and more frequent interaction loops for the human-robot collaboration.

3.3 Collaboration Style Based upon Mixed-Initiative Interactions

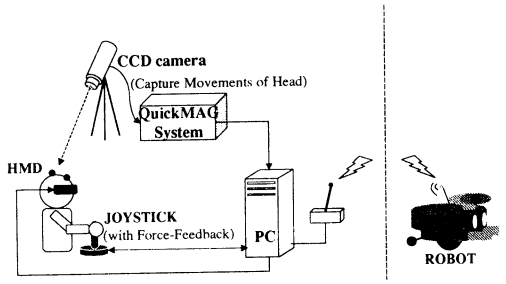
To realize the close human-robot interaction allowing their epistemic actions, we propose to add the constraints of their action depending on their environmental state. Figure 3 summarizes this proposal. In this model, there are mutual constraints between [PP,PA] variables for the operator and for the robot, which are enclosed by a broken line in the center of the figure. The operator's action to the system is restricted by the intention of the robot while the robot's action to the system is restricted by the intention of the operator. Because of these constraints, the interaction loops of both the operator and the robot may become smaller than the case without the constraints as shown in Figure 3.

On the other hand, the final behaviors of the robot itself only reflect the exchanges of the intentions of the operator and the robot's autonomy on the mutual constraints of variables. Hence, these constraints can be considered as their *shared communicational modalities* for their intent sharing. This style of interaction allows both of them to keep their independence and have the initiative to control the system depending on the situation (i.e., the mixed-initiative interactions).

4 Development of Robot Teleoperation System

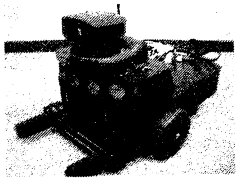
4.1 System Architecture

Figure 4 shows the architecture of the teleoperating environment we have developed. As a teleoperated robot, we adopt ActivMedia's PIONEER1 mobile robot, which has a CCD camera capable of panning, tilting and zooming (i.e., a PTZ camera), seven super sonic range sensors, and a gripper in front of its body.



(a)

(b)



(c)

Figure 4: Architecture of our robot teleoperation system: (a) a system overview; (b) a look of our system; and (c) PIONEER1 mobile robot

This robot can measure the distances from obstacles it by the super sonic range sensors, and can identify one target object with its image processing of the real image from the PTZ camera.

The operator manipulates the MICROSOFT Force Feedback Pro joystick with a mechanism to generate feedback force, controlling the robot on the basis of the display information through a HMD (or a head-mounted display). The display information consists of the real image from the PTZ camera and the virtual 3D space representing the robot's internal state. The operator's head movements are transformed into the movements of the PTZ camera, that are monitored by a motion capture device of QuickMAG system which traces the movements of two color markers attached on the top of the operator's head. This mapping enables the operator to deal with the operation of the PTZ camera intuitively.

4.2 Robot's Autonomy

We adopt a potential field method as the robot's autonomy producing the appropriate *velocity* and *steering* commands to navigate the robot. These two commands are generated from the potential field that is composed of repulsive forces from obstacles and an attractant force from a target object, to avoid obstacles and to move to the target object, both of which are detected by robot's machine-autonomy. Technical details of this robot autonomy was published elsewhere [2]. With this autonomy, the robot can perform the simple behaviors of the collision avoidance and the target pursuit without the operator's intervention.

4.3 Force-Feedback-Joystick as Shared Communicational Modality

The force-feedback-joystick in our teleoperation system plays the most significant part, that is the shared communicational modality we mentioned in subsection 3.3. The system is designed to enable the robot's autonomy to manipulate the stick using the feedback force proportional to its internal potential field as well as the operator.

The generated force can restrict the operator's input whose direction is opposite to it. On the other hand, the operator's resistance to this feedback does force the robot's autonomy to change the way to generate the potential field (i.e., the gain parameter of R). Hence, both the operator and the autonomy can mutually constraint each other's action.

In order to realize their mixed-initiative interactions, the velocity and steering commands to be commanded to the robot are given only by the input values of the forward-backward axis and the right-left axis of the stick respectively, that are the results of their interactions through the joystick.

4.4 Display Information

As the display information to the operator, the virtual 3D space that expresses the robot's internal state of its environment recognition are displayed through the HMD as well as the real image of the PTZ camera (Figure 5).

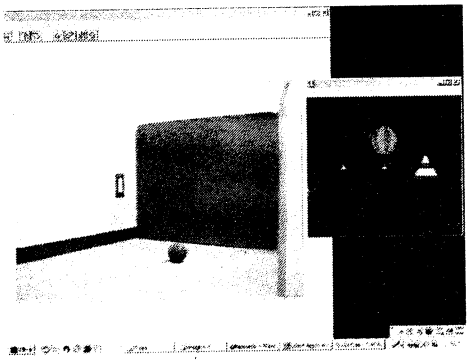


Figure 5: The display information.

5 Experiments

In order to evaluate our proposing system, we had experiments to compare its performance with the one of the conventional system in Figure 2 (i.e., with no force-feedback). The task is to navigate the robot at the corridor environment (Figure 6). This task is very hard without the support of the robot's autonomy as the PTZ camera has the broad blind area that makes the operator's recognition of the environment more difficult. In the experiments, a subject alternately performs the trial with "no-force-feedback" and with "force-feedback" five times for each.

As the results of these experiments, Table 1 shows the comparisons of performances in two different experimental settings. As for comparison, results of the

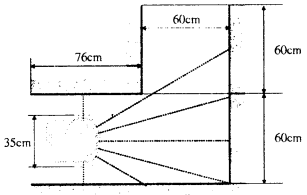


Figure 6: A corridor environment for experiments.

	MAN	without MII	with MII
$\bar{P}_{suc}$	0	0.4	0.4
$\bar{T}$ [sec]	53.60	32.36	25.41
$\bar{T}_S$ [sec]	–	18.88	17.25
$\bar{T}_F$ [sec]	53.60	41.35	30.85
$\bar{N}_{cw}$	1.2	3	1.33

Table 1: Comparisons of performances

trials made without any robot autonomy is also shown (i.e., MAN in the table). Wherein, $\bar{P}_{suc}$ denotes a success ratio of the trials, and $\bar{T}$ represents an averaged time for execution. $\bar{T}_S$ and $\bar{T}_F$ represent the averaged execution time in successful trials and in failed trials having collisions with the wall, respectively. $\bar{N}_{cw}$ shows the averaged number of motor stalls occurred in the task execution. All of these indexes show our proposing model with mixed initiative interactions has improved the task performances in comparison with the other two settings. These results can be considered as the successful sharing of the recognition of the situation between the operator and the robot.

Looking into the details of the experiment results, in case of no-force-feedback (No-FF) we could observe the operational errors due to fail in the awareness of the autonomy's intervention (i.e., the overfull operation causing the robot stuck into the wall). On the other hand, this problem is solved in our proposing system as shown in Figure 7, where the feedback force promotes the operator's awareness of the autonomy's intervention, and their confliction brings their mutual revision of actions, so that the appropriate control of the robot's behavior are generated.

6 Lens Model Analysis of Collaboration

6.1 Modified Usage of Lens Model

To quantitatively evaluate the effectiveness of our proposing method, we introduce Brunswik's lens model that is applied to the data gathered from the system in use. Although the original lens model analyzes the degree of coupling strength between the task environment and the human judge, we modify such a usage into the following. As for a construction of the ecological validity part of the model (i.e., right hand side of the model), we gathered an operation history made by a proficient operator with the robot autonomy off. Since such an

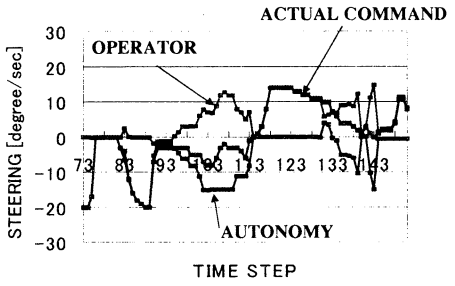


Figure 7: The improvement of the overfull operation

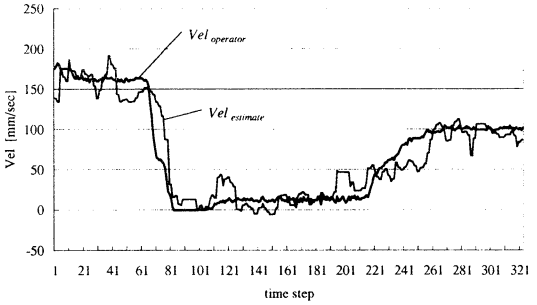


Figure 8: Profiles of operation commands of translational velocity with their linear estimates.

operator has acquired enough skills and do not need any aids from the machine autonomy, we can use the knowhows used by this expert operator as a reference model (i.e., a task model) for the lens model analysis. As for the cues, we used two variables of current status of the robot's translational velocity and rotational velocity in addition to seven variables each of which corresponds to the readings of ultrasonic rage sensors aligned in seven directions. Having regressed the linear model for two target variables of VEL and $RVEL$ from those data, we compared the outputs of the derived model with the actual operation commands as illustrated in Fig.8, which is for a variable VEL. This demonstrates that the regression model quite well explains the performance of the expert's.

Then, we built the left hand side of the lens model, in which the human-robot collaborative performance attained by our proposed architecture is evaluated. The human operator here is a less skilled operator than the former mentioned expert. As for comparison, we gathered two kinds of data; one kind of the data is obtained when our proposing architecture is used (with MII), and the other is the one obtained without our architecture in which the operator's awareness on the robot autonomy status is quite poor (without MII). Among the repetitive iterations of the trials, we classify them into two groups; one is a class of successful trials in which an operator does not let the robot collide with the wall at all, and the other one is a class of failed trials in which a robot collides with the wall and the operation is resumed after recovering from the

	O_{VEL}	O_{RVEL}
without MII	0.7370	0.8509
with MII	0.8969	0.8269

Table 2: Correlations between the linear models of the task and of *successful* operation profiles

	O_{VEL}	O_{RVEL}
without MII	0.7640	0.4297
with MII	0.7198	0.4478

Table 3: Correlations between the linear models of the task and of *failed* operation profiles

collisions. For the latter class of data, we excluded the subsequences of operations for recovery from the collisions out of the entire history, and the rests of the data were used for the analysis. The difference of the two classes is that the class of success corresponds to a history with well-trained situations for the operator, while the class of failure corresponds to a history with such less-trained, anomalous situations that may lead to the unexpected collisions with the wall.

For those four kinds of data (i.e., success/failed and with MII/without MII), we regressed the policies adopted by the human-machine collaborative systems as linear models relating among the cues and the system's output commands on VER and $RVEL$. Finally, we calculated the correlations with the outputs of the above-mentioned task model build from the expert and the outputs of the regressed policies. These results obtained from successful trials and from failed trials are shown in Table 2 and Table 3, respectively.

6.2 Discussions

We can conclude from the above results as follows. During the normal operations of success, the MII is contributing well to the total performance, which is verified from the fact that it reveals a higher correlation with the expert's performance with respect to the variable VEL. So the robot autonomy is a good support for the human autonomy. On the other hand, in the abnormal operations of failure, the MII is not contributing as expected, rather it degrades the collaborative performance (i.e., with a lower correlation with the expert's performance). This is a well-known phenomenon called *automation-induced surprises*.

The difference of the collaborative performance between two target variables VER and $RVEL$ can be interpreted as follows. Collaboration is going distinctively better for the target variable VER than $RVEL$. VER is a variable of translational movements, while $RVEL$ is a variable of rotational movements. In our experiment setting, an operator is allowed to use the live video stream from the on board PTZ camera that always captures the robot's front scenes, but does not capture the side views. With an aid of this, the operator can have good situation awareness on the status of the robot with respect to its translational movements, while he/she fails in keeping awareness on its rotational movements of the robot.

Although the robot autonomy is equivalently functioning both for the translational and rotational movements and force feedback from the joystick is obtained similarly, the restriction on the availability of visual stimulus is causing the difference of the operator's awareness on the status of the robot autonomy, which resulted in the difference of the collaborative performance. This is verified by the fact that the correlation values with the expert's are totally low for $RVEL$ than VEL. The difference between the collaboration performances with MII and without MII for the target variable VEL demonstrates a general conclusion that the collaboration between the human and the machine autonomies is not good if the status of the robot autonomy is not in aware by the human. This is another important aspect of *automation-induced surprises* caused by the black box of the machine autonomy.

7 Conclusions

We have developed a robot teleoperation system to provide the naturalistic collaboration style between a human operator and an autonomous robot, in which *shared communicational modalities* contribute to their intent sharing, allowing variability of their proactive actions and their mixed-initiative interactions. The lens model analysis was made for quantitatively evaluating the collaborative performance. From the results of experiments we also found that the existence of cues on the status of the machine autonomy other than the fed back force may affect on the performance of the collaboration (i.e., visual cues). To guarantee the usability of the teleoperation systems, we have to extend our analysis towards more formal analysis taking account of the factors of mediating ecological environments and towards evaluating degrees of intent-sharing between human- and robot-autonomies.

References

[1] R. W. Cooksey, *Judgment Analysis: Theory Methods and Applications*, Academic Press, 1996

[2] Y. Horiguchi, Tetsuo Sawaragi, Go Akashi, Naturalistic Human-Robot Collaboration Based upon Mixed-Initiative Interactions in Teleoperating Environment, Proc. of the 2000 IEEE International Conference on SMC, pp.876-881, 2000

[3] A. Kirlik, The Ecological Expert: Acting to Create Information to Guide Action, *Proceedings of Fourth Symposium on Human Interaction with Complex Systems (HICS'98)*, IEEE Press, 1998

[4] J. F. Allen, Mixed-Initiative Interaction, *IEEE Intelligent Systems*, IEEE Computer Society, 1999

[5] T. Sawaragi, T. Shiose, G. Akashi, Foundations for designing an ecological interface for mobile robot teleoperation, *Robotics and Autonomous Systems*, Vol. 31, pp.193–207, 2000.

[6] H. Ulrich, and G.J.B. Probst, *Self-Organization and Management of Social Systems: Insights, Promises, Doubts, and Questions*, Springer-Verlag GmbH & Co. KG, 1984.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Locomotion Control of a Multipod Locomotion Robot with CPG Principles

Katsuyoshi Tsujita*, Kazuo Tsuchiya*, Ahmet Onat†,
Shinya Aoi* and Manabu Kawakami*

* Dept. of Aeronautics and Astronautics
Kyoto University
Sakyo-ku, Kyoto 606-8501, Japan
† Sabanci University
Orhanli, 81474 Tuzla, Istanbul, Turkey

Abstract

This paper deals with a design of a control system for a multipod robot based on CPG principle. Oscillators are assigned at each leg and drive the periodic motion of legs. The phase of CPG is controlled by the signal of touch sensor which is mounted at the tip of the leg. It is confirmed through numerical simulation that the robot changes its gait pattern adaptively to variances of the environment.

1 Introduction

A walking robot is a robot with legs composed of links. Using the legs, the walking robot can move on a rough terrain and approach many locations. Then, research on a walking robot is proceeding actively[1]. Currently, control of locomotion of a walking robot is studied under the conditions that a desired gait pattern is given. At that time, the difficulty of control of a walking robot is to control of many elements according to the specified gait pattern. In the future, a walking robot is to carry out a task in the real world, where the geometric and kinematic conditions of the environment are not structured. At that time, the difficulty of control of a walking robot is not only to control of many elements according to the specified gait pattern but also to form a suitable gait pattern to a different circumstance. A walking robot is required to realize the real-time adaptability to a changing environment.

The walking motion of an animal seems to offer a solution to the problem; During a walking, a lot of joints and muscles are organized into a collective unit to be controlled as if it had fewer degrees of freedom but to retain the necessary flexibility for a changing environment[2]. It is widely believed that animal locomotion is generated and controlled, in part, by a central pattern generator (CPG)[3]. The CPG is a neuronal ensemble capable of producing rhythmic output in the absence of sensory feedback or brain input. The CPG, while not requiring external control for their basic operation, is highly sensitive to sensory feedback and external control from the brain. Sensory and descending systems are crucially involved in making the animal locomotion adaptive and stable.

A considerable amount of research has been done on design of a control system for walking robot which enables to adapt to variances of the environment based on the CPG principle[4~7]. M.A.Lewis et al developed a VLSI CPG Chip and using the chip, they implemented experiments of control of an underactuated running robotic leg; Periodic motion of the hip is driven by an oscillator, and then by controlling phase of oscillator using sensor signal, they established a stable running motion of the leg[4]. K.Akimoto et al designed a locomotion controller for hexapod robot by using CPG[5]. Oscillators, which are assigned for each leg, drive the periodic motion of each leg. The phase of oscillator is controlled by evaluating energy consumption of motors at joints of the legs. By using this control system, they realized a hexapod robot which can change the gait pattern adaptively to the walking velocity. The authors designed a control system for a quadruped robot by using CPG principle[6]. Oscillators, which are assigned for each leg, drive the periodic motion of each leg. The phase of oscillator is controlled by the signal of touch sensor at the tip of the leg. We confirmed through hardware experiment that the robot can walk stably by changing its gait pattern adaptively to variances of the environment. In this paper, a control system for a multipod robot based on CPG principle is proposed. Oscillators are assigned at each leg and

they drive the periodic motion of legs. The phase of oscillator is controlled by the signal of touch sensor at the tip of the leg. Through numerical simulation, it is confirmed that the robot changes its gait pattern adaptively to variances of the environment.

2 Equations of Motion

Consider the multipod robot shown in Fig. 1, which has five body modules and ten legs. Each leg is composed of two links which are connected to each other through a one degree of freedom (DOF) rotational joint. Each leg is connected to the body module through a one DOF rotational joint. The body modules are connected to each other through a two DOF rotational joint. The coordinate systems which are fixed at an inertial space and the first body module are defined as $[\boldsymbol{a}^{(-1)}] = [\boldsymbol{a}_1^{(-1)},\ \boldsymbol{a}_2^{(-1)},\ \boldsymbol{a}_3^{(-1)}]$ and $[\boldsymbol{a}^{(0)}] = [\boldsymbol{a}_1^{(0)},\ \boldsymbol{a}_2^{(0)},\ \boldsymbol{a}_3^{(0)}]$, respectively. Axes $\boldsymbol{a}_1^{(-1)}$ and $\boldsymbol{a}_3^{(-1)}$ coincide with the nominal direction of locomotion and vertically upward direction, respectively. Body modules are numbered from 1 to 5 and legs of each module are labeled as leg 1 for the left one and leg 2 for the right one, as shown in Fig. 1. The joints of each leg are numbered as joint 1 and 2 from the body module toward the tip of the leg. The position vector from the origin of $[\boldsymbol{a}^{(-1)}]$ to the origin of $[\boldsymbol{a}^{(0)}]$ is denoted by $\boldsymbol{r}^{(0)} = [\boldsymbol{a}^{(-1)}]r^{(0)}$. The angular velocity vector of $[\boldsymbol{a}^{(0)}]$ to $[\boldsymbol{a}^{(-1)}]$ is denoted by $\boldsymbol{\omega}^{(0)} = [\boldsymbol{a}^{(0)}]\omega^{(0)}$. We define $\theta_i^{(0)}$ $(i = 1,\ 2,\ 3)$ as the components of 1-2-3 Euler angle from $[\boldsymbol{a}^{(-1)}]$ to $[\boldsymbol{a}^{(0)}]$. We also define $\theta_k^{(i,j)}$ as the joint angle of link k of leg j of module i and $\theta_m^{(j)}(m = 1,2)$ as the angles between the body modules j and $j-1$.

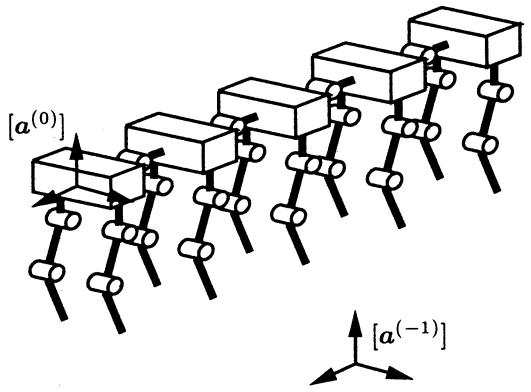


Fig. 1: Schematic model of a multipod robot

The state variable is defined as follows;

$$q^T = \left[\ r_m^{(0)} \quad \theta_m^{(0)} \quad \theta_l^{(j)} \quad \theta_l^{(i,k)}\ \right] \qquad (1)$$

$$i = 1,\cdots,5,\quad j = 2,\cdots,5,$$
$$k,l = 1,2,\quad m = 1,2,3$$

Equations of motion for state variable q are derived using Lagrange equations as follows;

$$M\ddot{q} + H(q,\ \dot{q}) = G + \sum(\tau_k^{(i,j)}) + \Lambda \qquad (2)$$

where M is the generalized mass matrix and $H(q,\ \dot{q})$ is the nonlinear term which includes Coriolis forces and centrifugal forces. G is the gravity term and $\sum(\tau_k^{(i,j)})$ is the input torque of the actuator at joint k of leg j of module i. Λ is the reaction force from the ground at the point where the tip of the leg makes contact. We assume that there is no slip between the tips of the legs and the ground.

3 Locomotion control

The architecture of the proposed control system is shown in Fig. 2; The control system is composed of leg motion controllers and a gait pattern controller. The leg motion controllers drive all the joint actuators of the legs so as to realize the desired motions that are generated by the gait pattern controller. The gait pattern controller involves non linear oscillators corresponding to each leg. The gait pattern controller receives the feedback signals from the touch sensors at the tips of the legs. A gait pattern emerges through modulation of the phases of the oscillators with the feedback signals from the touch sensors. The generated gait pattern is given to the leg motion controller as the commanded signal.

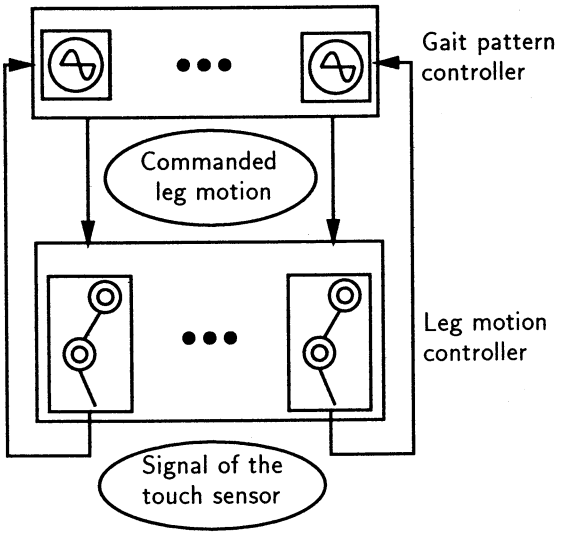


Fig. 2: Architecture of the proposed controller

3.1 Design of gait

Oscillator (i,k) is assigned on leg k of module i. The state of the oscillator (i,k) is expressed as follows;

$$z^{(i,k)} = \exp(j\ \phi^{(i,k)}) \tag{3}$$

where $z^{(i,k)}$ is a complex variable representing the state of the oscillator, $\phi^{(i,k)}$ is the phase of the oscillator and j is the imaginary unit.

We design the nominal trajectories of the tips of the legs as follows; We define the position of the tip of the leg where the transition from the swinging stage to the supporting stage as the anterior extreme position (AEP) and the position where the transition from the supporting stage to the swinging stage as the posterior extreme position (PEP) and then define the nominal PEP, $\hat{r}_{eP}^{(i,j)}$ and the nominal AEP, $\hat{r}_{eA}^{(i,j)}$ in the coordinate system $[\boldsymbol{a}^{(i)}]$ where the index $\hat{*}$ indicates the nominal value. We set the nominal trajectory for the swinging stage, $\hat{r}_{eF}^{(i,j)}$ as a closed curve which involves the points $\hat{r}_{eA}^{(i,j)}$ and $\hat{r}_{eP}^{(i,j)}$, and the nominal trajectory for the supporting stage, $\hat{r}_{eS}^{(i,j)}$ as a straight line which also involves the points $\hat{r}_{eA}^{(i,j)}$ and $\hat{r}_{eP}^{(i,j)}$. On the other hand, the nominal phase dynamics of the oscillator is defined as follows;

$$\dot{\hat{\phi}}^{(i,j)} = \omega \tag{4}$$

The nominal phases at AEP and PEP are determined as follows;

$$\hat{\phi}^{(i,j)} = \hat{\phi}_A^{(i,j)} \quad \text{at AEP}, \qquad \hat{\phi}^{(i,j)} = \hat{0} \quad \text{at PEP} \tag{5}$$

The nominal trajectories $\hat{r}_{eF}^{(i,j)}$ and $\hat{r}_{eS}^{(i,j)}$ are given as functions of the phase $\hat{\phi}^{(i,j)}$ of the oscillator as

$$\hat{r}_{eF}^{(i,j)} = \hat{r}_{eF}^{(i,j)}(\hat{\phi}^{(i,j)}) \tag{6}$$

$$\hat{r}_{eS}^{(i,j)} = \hat{r}_{eS}^{(i,j)}(\hat{\phi}^{(i,j)}) \tag{7}$$

Using these two trajectories alternatively we design the nominal trajectory of the tip of the leg $\hat{r}_e^{(i,j)}(\hat{\phi}^{(i,j)})$ as follows(Fig. 3);

$$\hat{r}_e^{(i,j)}(\hat{\phi}^{(i,j)}) = \begin{cases} \hat{r}_{eF}^{(i,j)}(\hat{\phi}^{(i,j)}) & 0 \le \hat{\phi}^{(i,j)} < \hat{\phi}_A^{(i,j)} \\ \hat{r}_{eS}^{(i,j)}(\hat{\phi}^{(i,j)}) & \hat{\phi}_A^{(i,j)} \le \hat{\phi}^{(i,j)} < 2\pi \end{cases} \tag{8}$$

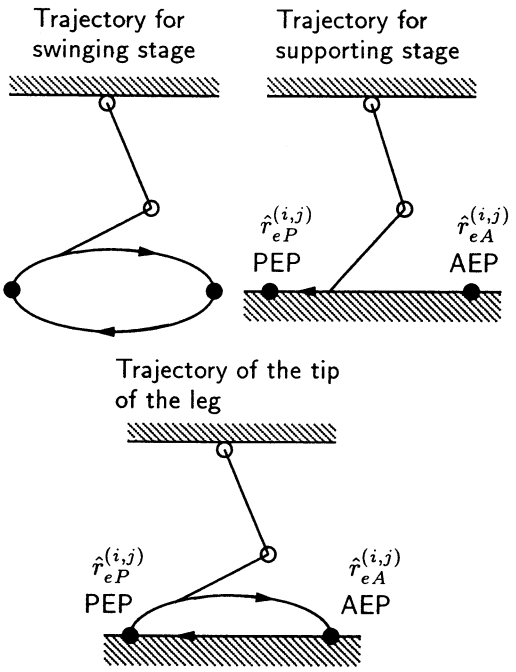


Fig. 3: Nominal trajectory of the tip of the leg

The nominal duty ratio $\hat{\beta}^{(i,j)}$ for leg j of module i is defined to represent the ratio between the nominal time for the supporting stage and the period of one cycle of the nominal locomotion.

$$\hat{\beta}^{(i,j)} = 1 - \frac{\hat{\phi}_A^{(i,j)}}{2\pi} \tag{9}$$

The nominal stride$\hat{S}^{(i,j)}$ of leg j of module i and the nominal locomotion velocity $\hat{v}$ are given as follows;

$$\hat{S}^{(i,j)} = \hat{r}_{eA}^{(i,j)} - \hat{r}_{eP}^{(i,j)}, \qquad \hat{v} = \frac{\hat{S}^{(i,j)}}{\hat{\beta}^{(i,j)}\hat{T}} \tag{10}$$

where, $\hat{T}$ is the nominal time period for a locomotion cycle.

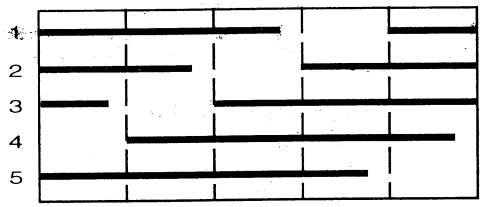
The gait patterns are defined as the relationships between motions of the legs. There are many gait patterns of the multipod robot. Suppose that the motion of legs of each module are in the same phase. One of the typical gait patterns is the pattern in which all of the phase relation between the legs of the neighboring two module are same (gait pattern #1). This pattern is called metachronal gait in the case of walking insect. In this pattern, the wave of swing stages moves from rear to front. Another typical gait pattern is a pattern in which some of the legs moves in the same phase (gait pattern #2). This pattern is called tripod gait in the case of walking insect.

Figure 4 shows the gait pattern diagrams of gait pattern #1 and #2 where the thick solid lines represent supporting stages. In general, each pattern is

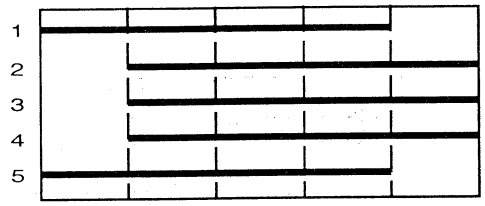
represented by a corresponding matrix of phase differences $\Gamma_{ii',jj'}$ as follows;

$$\phi^{(i',j')} = \phi^{(i,j)} + \Gamma_{ii',jj'} \quad (11)$$

where $\Gamma_{ii',jj'}$ is a phase difference of oscillator (i,j) and oscillator (i',j').



Gait pattern #1



Gait pattern #2

Fig. 4: Gait patterns

3.2 Control of gait

(i) Leg motion controller

The angle $\hat{\theta}_k^{(i,j)}$ of joint k of leg j of module i is derived from the trajectory $\hat{r}_e^{(i,j)}(\hat{\phi}^{(i,j)})$ and is written as a function of phase $\hat{\phi}^{(i,j)}$ as follows;

$$\hat{\theta}_k^{(i,j)} = \hat{\theta}_k^{(i,j)}(\hat{\phi}^{(i,j)}) \quad (12)$$

The commanded torque at each joint of the leg is obtained by using local PD feedback control as follows;

$$\tau_k^{(i,j)} = K_{Pk}^{(i,j)}(\hat{\theta}_k^{(i,j)} - \theta_k^{(i,j)}) + K_{Dk}^{(i,j)}(\dot{\hat{\theta}}_k^{(i,j)} - \dot{\theta}_k^{(i,j)}) \quad (13)$$
$$(i = 1, \cdots, 5, j, k = 1, 2)$$

where $\tau_k^{(i,j)}$ is the actuator torque at joint k of leg j of module i, and $K_{Pk}^{(i,j)}$, $K_{Dk}^{(i,j)}$ are the feedback gains, the values of which are common to all joints in all legs.

(ii) Gait pattern controller

We design the phase dynamics of oscillator i as follows;

$$\dot{\phi}^{(i,j)} = \omega + g^{(i,j)} \quad (i = 1, \cdots, 5, j = 1, 2) \quad (14)$$

where $g^{(i,j)}$ is the term caused by the feedback signal of the touch sensors of the legs.

Function $g^{(i,j)}$ is designed in the following way: Suppose that $\phi_A^{(i,j)}$ is the phase of leg i at the instant when leg i touches the ground. Similarly, $r_{eA}^{(i,j)}$ is the position of leg j of module i at that instance. When leg i touches the ground, the following procedure is undertaken.

1. Change the phase of the oscillator for leg j of module i from $\phi_A^{(i,j)}$ to $\hat{\phi}_A^{(i,j)}$.
2. Alter the nominal trajectory of the tip of leg i from the swinging trajectory $\hat{r}_{eF}^{(i,j)}$ to the supporting trajectory $\hat{r}_{eS}^{(i,j)}$.
3. Replace parameter $\hat{r}_{eA}^{(i,j)}$, that is one of the parameters of the nominal trajectory $\hat{r}_{eS}^{(i,j)}$, with $r_{eA}^{(i,j)}$.

Then, function $g^{(i,j)}$ is given as follows:

$$g^{(i,j)} = \hat{\phi}_A^{(i,j)} - \phi_A^{(i,j)} \quad (15)$$
at the instant leg j of module i touches the ground

As a result, the oscillators form a dynamic system that affect each other through the pulse-like interactions caused by the feedback signals from the touch sensor. Through the interaction, the oscillators generate gait patterns adaptive to the changing environment.

4 Numerical Analysis

Dynamic properties of the designed legged robot is investigated through numerical simulation. Purpose of the analysis is to verify that gait patterns adapted to the variances of the environment can emerge by using Eq. (14); That is to verify that oscillators which interact through only the feedback signals from the touch sensors at the tips of the legs can form a pattern of phase difference adapted to the variances of the environment. Physical parameters of the robot is shown in Table 1.

Table 1

Body module		
Width	0.13	[m]
Length	0.14	[m]
Height	0.08	[m]
Total Mass	8.0	[kg]
Legs		
Length of link 1	0.075	[m]
Length of link 2	0.075	[m]
Mass of link 1	0.20	[kg]
Mass of link 2	0.10	[kg]

Walking velocity (parameter β) is selected as a parameter of variance of the environment. Initial conditions are given as follows; A couple of oscillators on each module are in the same phase and the phase differences between every neighboring two modules are all same. Each value of the joint angle of the leg is determined by using the phase of the oscillator. All modules are in the statically steady states on a flat ground. In the simulation, because of the left-right symmetry, the robot has no roll motion and the legs of a module move in the same phase. Then, in the following, the suffix for leg is omitted. Simulation time is 250 steps. The results of the simulation are shown in Figs. $5 \sim 7$. Figure 5 shows the phase difference in a steady state of the oscillators, where $\Delta\Gamma_{i5}$ is a phase difference between the oscillators of module 5 and module i. Hatched area in the figure expresses the area where no steady state is obtained until the end of simulation time. Figure 6 shows the gait pattern diagram. Solid and blank lines express the supporting stage and the swinging stage, respectively. When the duty ratio β is large value, oscillators are divided into some groups in terms of the phase (gait pattern #2). For example, at $\beta = 0.8$, the oscillators are clustered into three groups, (2,3,4), (1) and (5). At $\beta = 0.75$, also three groups but another combination, (1,3), (2,4) and (5) is obtained. On the other hand, when β is small value, all the phase differences between the oscillators of neighboring two modules are in the same (gait pattern #1). For example, in the case of $\beta = 0.66$ this type of gait pattern is obtained. Figure 7 shows the time history of phase of the oscillator in Poincaré section. Poincaré section is selected at the timing when the phase of module 5, $\phi^{(5,j)}$ returns to the same value.

From the above results, it may be revealed that the phase pattern (gait pattern) change according to the variance of the value of β (the variance of walking velocity) and that there are some areas where no steady phase pattern is obtained between the different two phase patterns.

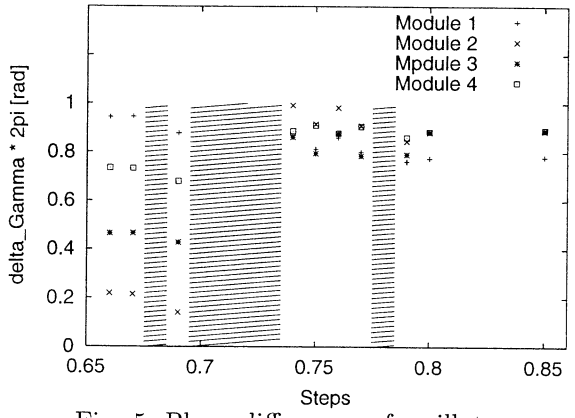
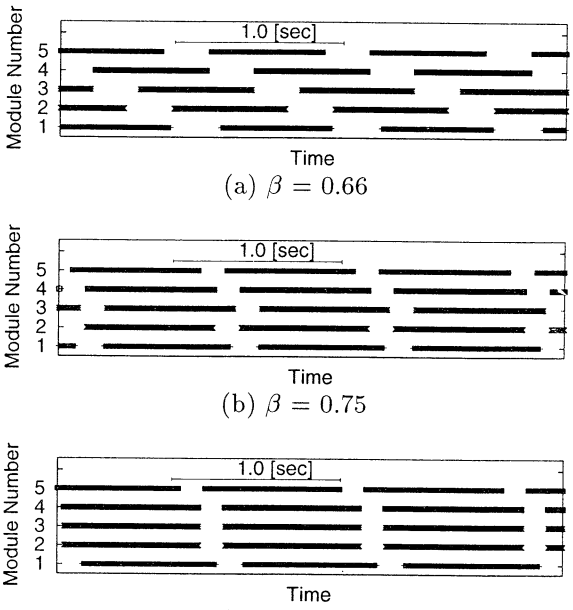


Fig. 5: Phase differences of oscillators

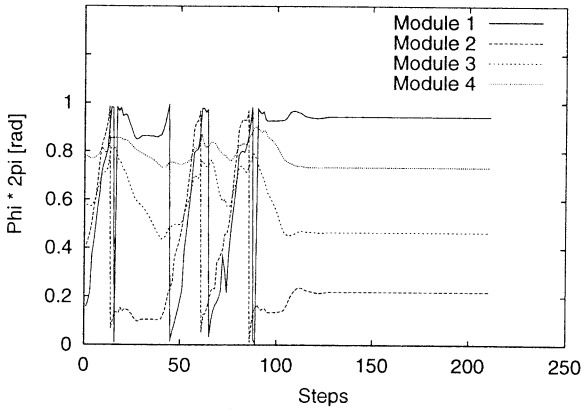


(a) $\beta = 0.66$

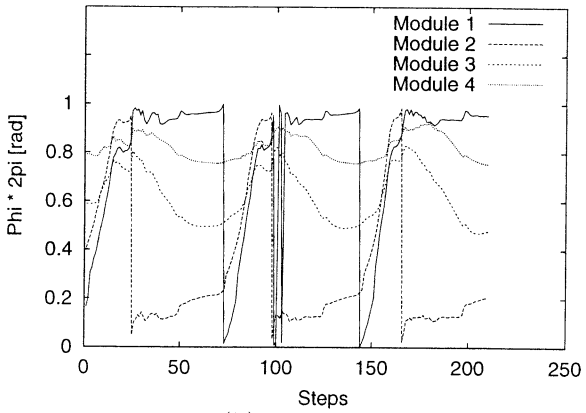
(b) $\beta = 0.75$

(c) $\beta = 0.80$

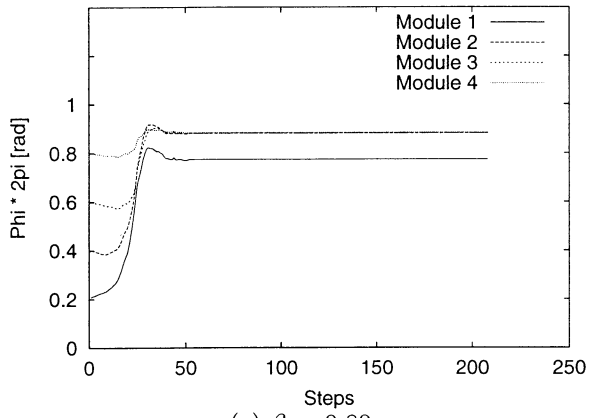
Fig. 6: Gait pattern diagram



(a) $\beta = 0.66$



(b) $\beta = 0.70$



(c) $\beta = 0.80$

Fig. 7: Time history of phase of the oscillator

5 Conclusions

In this paper, we proposed a controller of a multipod locomotion robot based on CPG principle. Oscillators are assigned at each leg and drive the periodic motion of legs. The phases of the oscillators are regulated impulsively by the feedback signals from the touch sensors at the tips of the legs. The time in which the tip of the leg of a body module touches the ground is a function of the positions and attitudes of other body modules. That is, this type of system of oscillators is the system of oscillators with impulsive mean field interactions. Numerically, this type of system is revealed to form phase patterns adaptively to the change of environment. But, the gait patterns emerged are somewhat sensitive to the variations of the values of the parameters. In order to improve the stability of the system, the mutual interaction derived from a certain potential function is to be added to the system. The design of such interactions remains for a future work.

Acknowledgments

The authors were funded by grants from the The Japan Society for the Promotion of Science (JSPS) as the Research for the Future program (RFTF) and from the Japan Science and Technology Corporation (JST) as the Core Research for Evolutional Science and Technology program (CREST).

References

1. Int. J. Robotics Research Vol.3, No.2,1984

2. N.A.Bernstein (1967) Co-ordination and regulation of movements. Oxford, Pergamon press, New York

3. S.Grillner (1985) Neurobiological Bases of Rhythmic Motor Acts in Vertebrates. Science Vol.228, pp. 143-149

4. M.A.Lewis, R.E.Cummmings, A.H.Cohen and M.Hartmann (2000) Toward Biomorphic Control Using Custom a VLSI CPG Chips. Proc. of International Conference on Robotics and Automation 2000

5. K. Akimoto, S. Watanabe and M. Yano (1999) An insect robot controlled by emergence of gait patterns. Proc. of International Symposium on Artificial Life and Robotics, Vol. 3, No. 2, pp. 102-105

6. K. Tsujita, K. Tsuchiya and A. Onat (2000) Decentralized Autonomous Control of a Quadruped Locomotion Robot Proc. of AMAM 2000, E-18

7. H. Kimura, K. Sakaura and S. Akiyama (1998) Dynamic Walking and Running of the Quadruped Using Neural Oscillator. Proc. of IROS'98, Vol. 1, pp. 50-57

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Method for Evaluating Thermo Dynamical and Statistical Molecular Dynamical Interactions among the Bio Molecular Particles that are Participating for Gene Expression.

H. Hirayama and *Y Okita

Department of Public Health Asahikawa Medical College
E mail hirayama@asahikawa-med.ac.jp * Shizuoka university.

Summary

A mathematical method was introduced for evaluating the bio molecular dynamical interaction in terms of the gene expression and regulation of an artificial life. We introduced minutely the mathematical processes for shear viscosity, thermal conductivity of interacting bio molecules that have different geometries, number density and mass.
key words. Gene expression, Molecular thermo dynamics,

1. Introduction.

Regulating the expression of genetic function is the first step for creating an artificial life. Activation process of genes is determined by the interactions among the regulating protein particles, inducer particles that counter act with the repressor and target bases of the special region on the genes [1],[2],[3]. The ultimate process of the gene expression can be described by physical collisional encounters of these bio molecular particles.

We introduce a mathematical method [4][5],[6] that was originally directed for molecular thermo dynamics of dilute Gas. We show computed results for shear viscosity and thermal conductivity as functions of mass of interacting molecules, number density for two interacting bio molecules which have mutually different geometries.

2. Mathematical expansion.

2-1. Boltzmann equation and collision integral.

The distribution functions [4] of are defined such that

$fi(\mathbf{r}, \mathbf{v}_i, \alpha_i, \omega_i, t)\, d\mathbf{r}\, d\mathbf{v}_i\, d\alpha_i\, d\omega_i$ ----(1-1)

is the number of molecules of kind i which spatial position, linear velocity, orientation and angular velocity at tine t are in the hypervolume element $d\mathbf{r}\, d\mathbf{v}_i\, d\alpha_i\, d\omega_i$ Using the Taylor series expansion [4], the relation to fi with Ji (the net rate of joining in the volume element $d\mathbf{r}$ $d\mathbf{v}_i\, d\alpha_i\, d\omega_i$ due to the collision of molecules i

$$D_i f_i = J_i = \Sigma J_{ij} \quad \text{---(1-2)} \quad \text{where}$$

$$D_i = \partial/\partial t + \partial/\partial \mathbf{r} \cdot \mathbf{v}_i + \partial/\partial \mathbf{v}_i \cdot (\mathbf{F}_i/m_i) + \partial/\partial \alpha_i \cdot \partial \alpha_i/\partial t + \partial/\partial \omega_i \cdot \partial \omega_i/\partial t \quad \text{----(1-3)}$$

$\mathbf{F}_i$ is the external force which mass is m_i. A loss of molecules of kind i is given by [4]

$-f_i f_j (\mathbf{k} \cdot \mathbf{g}_{ji})\, S(\mathbf{k})\, d\mathbf{k}\, d\mathbf{r} dt\, d\mathbf{v}_i\, d\alpha_i\, d\omega_i\, d\mathbf{v}_j\, d\alpha_j\, d\omega_j$

where $\mathbf{g}_{ji}$: the relative velocity of the points of contact, before the encounter $\mathbf{g}_{ji} = \mathbf{v}_j - \mathbf{v}_i + (\omega_j \times \sigma_j) - (\omega_i \times \sigma_i)$ σ_i : the radius vector from the center of a molecule i to the point of contact. $\mathbf{k}$: a unit normal vector to the tangential plane of the contact of the collision which directs from molecule i to molecule j. $S(\mathbf{k})\, d\mathbf{k}$: the element of surface of the volume from which the center of molecule j is excluded when the molecules are in contact. By using the pseudo-inverse collisions, denoting ' as the quantities before the collision, we obtain

$$J_{ij} = \int (f_i' f_j' - f_i f_j)(\mathbf{k} \cdot \mathbf{g}_{ji}) S(\mathbf{k})\, d\mathbf{k}\, d\mathbf{v}_j\, d\alpha_j\, d\omega_j$$

2-2. Solution of the Boltzmann equations.

The equilibrium solutions f_i^0 for the Boltzmann equations

$$D f_i^0/f_i^0 = [-1/3 (m_i V_i^2/(2kT) + \Omega_i \cdot \mathbf{I}_i \cdot \Omega_i/(2kT)) U + m_i \mathbf{V}_i \mathbf{V}_i/(kT)] : \partial \mathbf{v}_o/\partial \mathbf{r} + \mathbf{I}_i \cdot \Omega_i \cdot \mathbf{V}_i/(kT) : \partial \omega_o/\partial \mathbf{r} + n/n_i^* (\mathbf{e}_i \cdot \Omega_i + \mathbf{d}_i \cdot \mathbf{V}_i) - (4 - m_i V_i^2/(2kT) - \Omega_i \cdot \mathbf{I}_i \cdot \Omega_i/(2kT)) \mathbf{V}_i \cdot \partial \log T/\partial \mathbf{r}$$

where $V_i = v_i - v_o$ is the velocity relative to the local stream velocity v_o, $\Omega_i = \omega_i - \omega_o$ is the angular velocity relative to the local average angular velocity ω_o, $\mathbf{I}_i$: the moment of inertia tensor of molecule i in the space fixed coordinate system. k :Boltzmann coefficient. T: absolute temperature.

$$\mathbf{d}_i = \partial/\partial \mathbf{r}\, (n_i^*/n_i) + n_i^*/n_i (1 - m_i/m)\, \partial \log p/\partial \mathbf{r} - n_i^*/(n_i kT)(\mathbf{F}_i - m_i/m_o \mathbf{F}_i^0).$$

$\mathbf{e}_i = -n_i^*/(n\,kT)[\mathbf{G}_i - \mathbf{I}_i \cdot (\mathbf{I}^0)^{-1} \cdot \mathbf{G}_i^0]$ where $\mathbf{G}_i$ is the torque on molecule i. : denotes vector dyadic.

Expanding the distribution functions in power series by a perturbation parameter Writing [4] $f_i = f_i^{(0)}(1 + \Phi_i)$

$$\phi_i = A_i^1 \cdot \partial \log T/\partial \mathbf{r} + \mathbb{A}_i^2 : \partial \mathbf{v}_o/\partial \mathbf{r} + \mathbb{A}_i^3 : \partial \omega_o/\partial \mathbf{r} + \Sigma (A_i^{4j} \cdot \mathbf{d}_j + A_i^{5j} \cdot \mathbf{e}_j) \quad \text{----(2-1)}$$

where A_i^1, A_i^{4j} A_i^{5j} are vectors, $\mathbb{A}_i^2$ $\mathbb{A}_i^3$ are tensors.

Fig 1

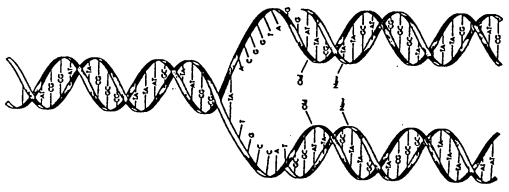
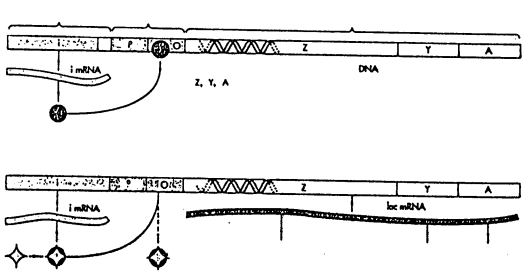


Fig 2



2-3. Determination of coefficients of solutions.

The scalar functions $A_{i,k}^{\nu}$ in (2-1) are approximated by finite series of products of Sonine polynomials. [5],[6].

$$A_i^{\nu} = \sum_{r'}^{N'} \sum_{r''}^{N''} a_{i;r'r''}^{\nu} S_{i;r'r''}^{\nu} \quad \text{-----(3-1)}$$

Application of the variational method [6] gives a set of coupled linear algebraic equations for the expansion coefficients $a_{i;r'r''}^{\nu}$. In reducing the integral equation to a set of algebraic equations, we define [5],[7]

$$R_{i;r'r''}^{\nu} = \int S_{i;r'r''}^{\nu} \mathbb{W}_i^{T} : \mathbb{K}_i^{\nu} \, dv_i \, d\omega_i \quad \text{--------(3-2)}$$

$$Q_{ij;\,r'r''m'm''}^{\nu} = \sum_p n_p^* \{ \delta_{ij} (\mathbb{W}_i^{\nu} S_{i;r'r''}^{\nu} ; \mathbb{W}_i^{\nu} S_{i;m'm''}^{\nu})_{ip} + \delta_{jp} (\mathbb{W}_i^{\nu} S_{i;r'r''}^{\nu} ; \mathbb{W}_i^{\nu} S_{i;m'm''}^{\nu})_{ip} \} \quad \text{----(3-3)}$$

where the definition of bracket and parenthesis integral operators are given in 4-4. By applying the variational method [6] , we arrive

$$R_{i;r'r''}^{\nu}(\alpha i) = - \sum_{j,m'm''} \sum \sum Q_{ij;\,r'r''m'm''}^{\nu}(\alpha i, \alpha j) \, a_{j;m'm''}^{\nu}(\alpha j) \, d\alpha j \quad \text{----(3-4)}$$

Equations (3-2) to (3-4) determine the best values of the constants $a_{j,;m'm''}^{\nu}$. From the orthogonality of the representation coefficients $D^{L}(\alpha i)_{\mu,s}$,

$$R_i^{\nu}(L,\mu,s\,|\,r'r'') = ((2L+1)/8\pi^2) \int R_{i;r'r''}^{\nu}(\alpha i) \, {}^{*}D^{L*}(\alpha i)_{\mu,s} \, d\alpha i \quad \text{---(3-5a)}$$

$$Q_{ij}^{\nu}(L,\mu,s,L',\mu',s'\,|\,r'r''m'm'') = ((2L+1)(2L'+1)/64\pi^2) \int \int n \sin\beta' / (n^*(\alpha i)') \, Q_{ij;\,r'r''m'm''}^{\nu} \, D^{L}(\alpha i)_{\mu,s} \, D^{L'}(\alpha j)_{\mu',s'}^{*} \, d\alpha i \, d\alpha i' \quad \text{---(3-5b)}$$

2-4-a. Bio molecular transport coefficients.

2-4a-1. Viscosity.

From the definition of pressure flux in thermodynamics

$$p = nkTU - 2\eta [1/2 [\partial v_0 / \partial r + (\partial v_0 / \partial r)^{T}] - 1/3 U (\partial / \partial r) \cdot v_0] - \kappa U \, \partial / \partial r \cdot v_0 \quad \text{--(4-1b)}$$

where η and κ are shear and bulk viscosities [6],[7],

$$\eta = -1/15 \sum m_i \int f_i^0 A_i^{21} V_i^4 \, dv_i \, d\omega_i \quad \text{----(4-1)}$$

Substituting (3-1), perform the integration over $dv_i \, d\omega_i$ and utilizing the orthogonal properties of the Sonine polynomials and $D^{L}(\alpha i)_{\mu,s}$, *shear* viscosities are

$$\eta = - 8\pi^2 (kT)^2 \sum n_i / m_i \, a_i^{21} (000\,|\,00) \quad \text{----(4-2)}$$

2-4a-2. Thermal conductivity

By the thermodynamical consideration [6],[7] we have the molecular heat flux as

$$q = - \lambda \, \partial T / \partial r + q(r) \quad \text{----(4-5a)}$$ where

$$\lambda = 4 \pi^2 k^2 T \sum n_i / m_i [5 (a_i^{11} (000\,|\,10) + a_i^{12} (000\,|\,10)) + 3 (a_i^{11} (000\,|\,01) + a_i^{12} (000\,|\,01))] \quad \text{----(4-3)}$$

is the thermal conductivity .

2-4b. Solution of the algebraic equations.

This chapter introduces a method for $a_j^{\nu}(L,\mu,s\,|\,m'm'')$ by solving algebraic equation (3-8). We introduce, only the methods for shear viscosity and thermal conductivity. First, we define

1. Bracket and parenthesis integrals [4],[6]

$$[G_i ; H_j]_{ij} = 1/(n_i^* n_j^*) \int G_i^{T} : [H_j' - H_j] f_i^0 f_j^0 (k \cdot g_{ji}) S(k) \, dk \, dv_i \, d\omega_i \, dv_j \, d\omega_j \quad \text{----(4-4)}$$

1. shear viscosity η

$$a_1^{21}(000\,|\,00) = 5 m_i / ((8 n_1 \pi^2) [n_2/n_1 \{ \mathbb{W}_1 ; \mathbb{W}_2 \}_{12} - \{ \mathbb{W}_2 ; \mathbb{W}_2 \}_{22}] / [\{ \mathbb{W}_1 ; \mathbb{W}_1 \}_{11} \{ \mathbb{W}_2 ; \mathbb{W}_2 \}_{22} - \{ \mathbb{W}_1 ; \mathbb{W}_2 \}_{12} \{ \mathbb{W}_2 ; \mathbb{W}_1 \}_{21}] \quad \text{------(4-7)}$$

Substituting this to (4-3), we obtain η. For thermal conductivity, we have the thermal conductivity by [7]

$$\lambda = 3 k^3 T^3 (5 \Pi f + 3 \Pi g + 5 \Pi h + 3 \Pi i + 5 \Pi j + 3 \Pi k + 5 \Pi l + 3 \Pi m) / (4 \Delta 2) \quad \text{-----(4-5)}$$

where $\Delta 2$ is the determinant [7] which elements have the form of brace integral such as $\{ V_i \, S_{1/2}^{1} (\varepsilon_i^{r}) ; S_{3/2}^{1} (\varepsilon_i^{t}) V_i \}_{ii}$. The determinants Πd--Πm are obtained from $\Delta 2$ by replacing columns as indicated in [7].

2-5. Evaluation of the Collision integrals.

2-5 -1. Linear velocities

The dimensionless relative linear velocity, [5],[7]

$$\mathbf{W_k} = (m_k / 2kT)^{1/2} \mathbf{V_k} \quad k=i, j$$

The relations between those before (denoted by ') and after the collision are

$$\mathbf{W_k'} = \mathbf{W_k} \pm M_k^{-1/2} \kappa \quad k=i + \text{ and } k=j -$$

$M_i = m_i / m_o \qquad M_j = m_j / m_o \qquad m_o = m_i + m_j$

κ associates with the change of linear momentum of one of the molecule during the collision

$$\kappa = 2 \xi_{ij} (\mathbf{k} \cdot \Gamma) (M_i M_j)^{1/2} \mathbf{k}$$

$$\xi_{ij}^{-1} = 1 + m_i m_j / m_o [(\sigma_i \times \mathbf{k}) \cdot \mathbf{I}_i^{-1} \cdot (\sigma_i \times \mathbf{k}) + (\sigma_j \times \mathbf{k}) \cdot \mathbf{I}_j^{-1} \cdot (\sigma_j \times \mathbf{k})] \quad \text{--(5-1)}$$

$$\Gamma = (\mu_{ij} / 2kT)^{1/2} (\mathbf{g}_{ji} - ((\omega_i \times \sigma_i) - (\omega_j \times \sigma_j)) \quad \text{---(5-2)}$$

is the dimensionless relative velocity of contact point of the collision. The dimensionless relative velocity of the colliding molecules, velocity of j relative to i is

$$\gamma = (\mu_{ij} / 2kT)^{1/2} \mathbf{g}_{ji} = (\mu_{ij} / 2kT)^{1/2} (\mathbf{V_j} - \mathbf{V_i})$$

$$\gamma_0 = (\mu_{ij} / 2kT)^{1/2} \mathbf{g}_0 \quad \text{where}$$

$\mu_{ij} = m_i m_j / m_o$ and $\mathbf{g}_0 = (\omega_i \times \sigma_i) - (\omega_j \times \sigma_j)$

2-5-2. Angular velocities and moments.

The dimensionless relative angular velocity is [5],[7]

$$\mathbf{w_i} = (\Gamma_1 \Gamma_2 \Gamma_3)^{1/6} \Omega_i / (2kT)^{1/2}$$

where $\Gamma_1 \Gamma_2 \Gamma_3$: the principal moments of inertia of a molecule. The relations of angular momenta before (denoted by ') and after the collisions are

$$\mathbf{I_i} \cdot \mathbf{w_i'} = \mathbf{I_i} \cdot \mathbf{w_i} \pm (\sigma_i \times \mathbf{k}) m_o^{1/2} \Gamma_i^{1/6}$$

k=i for + and k=j for -

2-5-3. Kinetic energies.

Relative translational and rotational kinetic energies

$$\varepsilon_i^{t} = m_i V_i^2 / (2kT) \qquad \varepsilon_i^{r} = \Omega_i \cdot \mathbf{I_i} \cdot \Omega_i / (2kT)$$

The dimensionless translational and rotational kinetic energies before a collision (denoted by ') are

$$\varepsilon_k^{(r)'} = \varepsilon_k^{(r)} \pm 2 m_o / \Gamma_k^{1/6} \mathbf{w_k} \cdot [\sigma_k \times \kappa] + m_o [\sigma_k \times \kappa] \cdot \mathbf{I_k^{-1}} \cdot [\sigma_k \times \kappa] \quad \text{-(5-3a)}$$

$$\varepsilon_k^{(t)'} = \varepsilon_k^{(t)} + M_k^{-1} \kappa^2 \pm 2 M_k^{-1} \kappa \cdot \mathbf{W_k} \quad : \quad k=i \text{ for + and } k=j \text{ for -} \quad \text{----(5-3b)}$$

The reduced velocity of the center of mass of the pair of colliding molecules.

$$\mathbf{G} = (m_o / 2kT)^{1/2} [M_i \mathbf{V_i} + M_j \mathbf{V_j}]$$

2-5-4. An example of collision integral

For first we evaluate the bracket integral $[\mathbf{V_i} ; \mathbf{V_j}]_{ij}$ which contributes to the brace expression { } leading to the diffusion coefficient.

$$[\mathbf{V_i} ; \mathbf{V_j}]_{ij} = -1/(2 n_{i,1}^* n_{j,k}^*) \int [(V_i' - V_i) (V_j' - V_j)]$$

$f_i^{(0)} f_j^{(0)} (\mathbf{k} \cdot \mathbf{g}_{ji}) S(\mathbf{k}) \, d\mathbf{k} \, dv_i \, dv_j \, d\omega_i \, d\omega_j$

Introducing Γ, $\mathbf{g}_{ji}$ and f_i^0 and change variables of from $(v_i, \omega_i, v_j, \omega_j)$ to (G, γ, w_i, w_2).

$$[V_i ; V_j]_{ij} = (2kT)^{3/2} / (2(\mu_{ij})^{3/2} \pi^6) \int\int\int\int\int \kappa^2 (\mathbf{k} \cdot \Gamma) \exp(-G^2 - \gamma^2 - \varepsilon_i^r - \varepsilon_j^r) S(\mathbf{k}) \, d\mathbf{k} \, dG \, d\gamma \, dw_i \, dw_j \quad \text{-------(5-4)}$$

After integrating over G

$$[V_i ; V_j]_{ij} = 2 (2kT)^{3/2} / ((m_i m_j m_o)^{1/2} \pi^{9/2}) \int \xi^2 (J_{30}^0 - 3 J_{21}^0 + 3 J_{12}^0 - J_{03}^0) S(\mathbf{k}) \, d\mathbf{k} \quad \text{--------(5-5)}$$

where, we define [8],

$$J_{\nu\nu'\zeta} = \pi \int\int d\gamma \, \gamma^{2\zeta} (\mathbf{k} \cdot \gamma)^{\nu} (\mathbf{k} \cdot \gamma_0)^{\nu'} \exp(-\gamma^2 - \varepsilon_i^r - \varepsilon_j^r) dw_i \, dw_j \quad \text{--(5-6)}$$

Then,

$$J_{\nu\nu'}^{(0)} = \pi^3 \eta^{\nu'} \int_u^{\eta} x^{\nu} u^{\nu'} \exp(-x^2 - u^2 - v^2) \, dx \, du \, dv$$

After substituting these and integrating over v, we change the variables from (x , u) to (r , θ) of polar coordinates by setting $x = r \cos\theta$ and $y = r \sin\theta$.

$$J_{\nu\nu'}^{(0)} = \pi^{7/2} / 2 \Gamma((\nu + \nu' + 2) / 2) \int d\theta \cos^{\nu}\theta \sin^{\nu'}\theta$$

Thus,we have reduced the brackets integration to

$$[V_i ; V_j]_{ij} = (2kT)^{3/2} / (\mu_i^{1/2} m_o \pi^{1/2}) \int \xi_{ij}^{1/2} S(\mathbf{k}) \, d\mathbf{k} \quad \text{--------(5-7)}$$

2-6. Reduction of the brace integral.

We proceed to evaluate the brace integral expressions $\{ \}_{ij}$ by the integrals over two orientations and over $\mathbf{k}$. For an example we analyze

$$n_1 \{ V_1 ; V_1 \}_{11} - n_2 \{ V_1 ; V_2 \}_{12} \quad \text{----(6-1)}$$

which participates for diffusion coefficient. Substituting the brackets and parenthesis integrations (5-33a,5-33b) into the brace expression (4-6-a) , (6-1) becomes

$$\text{(6-1)} = (2kT)^{3/2} \rho M_2 / ((8\pi^2)^2 \pi^{1/2} m_o \mu_{i,l}^{3/2}) \int \xi_{12}^{1/2} \sin\beta_1 \sin\beta_2 \, d\alpha_1 \, d\alpha_2 \, S(\mathbf{k}) \, d\mathbf{k} \quad \text{----- (6-2)}$$

To perform the integration for orientation space, variables were changed from the set of

$(\theta, \phi, \alpha_1, \beta_1, \gamma_1)$ to the set of $(\theta_1', \phi_1', \theta_2', \phi_2', \psi)$

which Jacobian is $-\sin\theta \sin\beta_1 / (\sin\theta_1' \sin\theta_2')$ where θ, ϕ are the angle variables of vector $\mathbf{k}$. Finally the integration is performed with respect to S(k). S(k) is the supporting functions [8],[9]. The supporting functions in turn can be expressed in terms of the principal radii of curvature of the surface of the colliding molecules.

$$\text{(6-1)} = (2kT)^{3/2} \rho M_2 / (4 m_o (\pi \mu_i^{3/2})) \int\int \xi_{ij}^{1/2} \, dS_i \, dS_j \left[1 / (q_1^{(i)} q_2^{(i)}) + 1 / (q_1^{(j)} q_2^{(j)}) + 1/2 \left(1 / q_1^{(i)} + 1 / q_2^{(i)} \right) \left(1 / q_1^{(j)} + 1 / q_2^{(j)} \right) \right] \quad \text{--(6-3)}$$

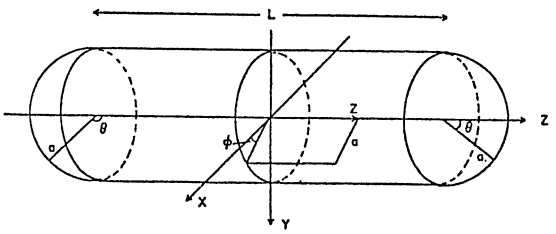
where dSi and dSj are elements of the surface of molecules i and j. $q_\lambda^{(i)}$ (λ =1 or 2) are two principal radii of curvature of the surface of molecule. Using the supporting function, we can relate ξ_{ij}, η_i, η_j to the principal radii of curvature of the surfaces of the models. Integration of aij can be carried out to reach for an example,

$$n_1 \{ V_1 ; V_1 \}_{11} - n_2 \{ V_1 ; V_2 \}_{12} = A_{12}(1) (2kT)^{3/2} \rho M_2 / (m_o \mu_{ij}^{3/2} \pi^{1/2})$$

2-7. Sphero-cylindrical particle

For practical application of the present method to gene regulation, we introduce coordinate system for sphero-cylindrical particles [7]. Fig 3 shows sphero cylindrical model and the coordinates characterizing its surface. The

Fig 3



cylinder is of length L and radius a. A point on the cylinder is described by the coordinate z, $-L/2 \leqq z \leqq L/2$. The azimuthal angle ranges $0 \leqq \phi \leqq 2\pi$. A point on the hemispherical caps is described by the polar angle $0 \leqq \theta \leqq \pi/2$ on the upper cap and $\pi/2 \leqq \theta \leqq \pi$ on the lower cap. Its azimuthal angle ranges $0 \leqq \phi \leqq 2\pi$.

The radius vector to a point on the surface in the body fixed coordinate system is for particle i

$\sigma_i = (a_i \cos\phi_i, \; a_i \sin\phi_i, \; z_i)$ ------(7-1a) for cylindrical section and

$\sigma_i = [a_i \sin\theta_i \cos\phi_i, \; a_i \sin\theta_i \sin\phi_i, \; \pm(L_i/2) + a_i \cos\theta_i]$ --(7-1b) for spherical caps

The unit vector normal to the surface is

$\mathbf{k} = (\cos\phi_i , \sin\phi_i , 0)$ -----(7-2a) for cylindrical section and

$\mathbf{k} = (\cos\phi_i \sin\theta_i , \sin\phi_i \sin\theta_i , \cos\theta_i)$ ---(7-2b) for spherical caps

Thus, the vectorial products are

$(\sigma_i \times \mathbf{k}) = (-z_i \sin\phi_i, \; z_i \cos\phi_i, \; 0)$ --(7-3a) for cylindrical section

$(\sigma_i \times \mathbf{k}) = [-\pm(L_i/2) \sin\theta_i \sin\phi_i, \; \pm(L_i/2) \sin\theta_i \cos\phi_i, \; 0]$ --(7-3-b) for spherical caps.

Setting that the geometrical symmetry axis is a principal axis of the mass distribution and $\Gamma_{i1} = \Gamma_{i2} = \Gamma_i$, we have following relations

$m_i \eta_i^2 / [2 (\Gamma_{i1} \Gamma_{i2} \Gamma_{i3})^{1/3}] = 4 \alpha_i z_i^2 / L_i^2$ --(7-4a) for cylindrical section

$= \alpha_i \sin^2\theta_i$ ---(7-4b) for spherical caps

where $\alpha_i = m_i L_i^2 / (8\Gamma_i)$ .Defining the geometric asymmetry parameter $\beta_i = L_i/(2a_i)$ and mass distribution parameter $\delta_i = \beta_i / (2\alpha_i)$,we can compute viscosity, diffusion coefficient and thermal conductivity by setting L_i, a_i and δ_i. The principal radii of curvature of the surface are

$q_1^i = a_i \quad q_2^i = oo$ for cylindrical section and $q_1^i = q_2^i = a_i$ for spherical caps.

The surface elements are $dS_i = a_i \, dz_i \, d\phi_i$ ---(7-5a) on the cylinder

$dS_i = a_i^2 \sin\theta_i \, d\theta_i \, d\phi_i$ -----(7-5-b) on the caps,

Applying these parameters for (6-3), integrating over ϕ_i, we can acheve integrations appearing in 6-8a, 6-8b and 6-8c in the brace integrals constituting the coefficients for viscosity , diffusion and thermal conductivity. [7]

3. Computational results.

To examine the influences of geometric properties of interacting two bio molecules, we set (n_1, n_2) = (1, 0.5), (m_1, m_2) = (1, 0.5), (L_1, L_2) = (1, 0.5), (a_1, a_2) = (1, 0.5) and (δ_1, δ_2) = (1, 0.5).

Fig 4-a shows three dimensional expression of normalized shear viscosity $\eta/(5kT/2)$ as functions of mass (m_2= 0.1, 0.3, 0.5, 0.7) and number density (n_2 = 0.1 to 0.9) of the second particle. Fig 4-b shows those as functions of radius (a_2 = 0.1 to 0.9) and length (L_2

Fig 4-a

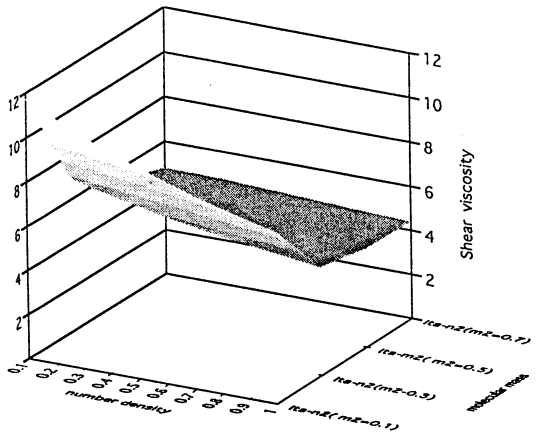


Fig 5-a

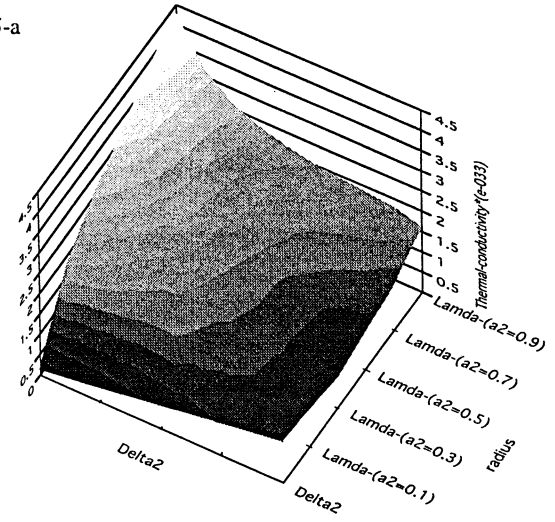


Fig 4-b

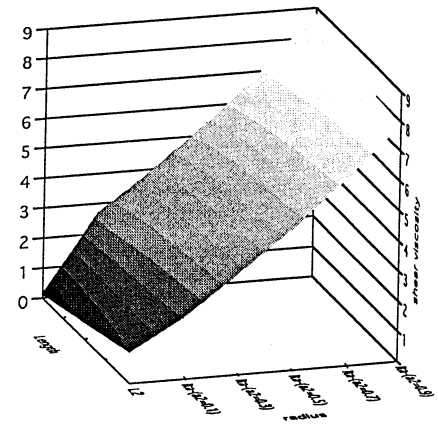


Fig 5-b

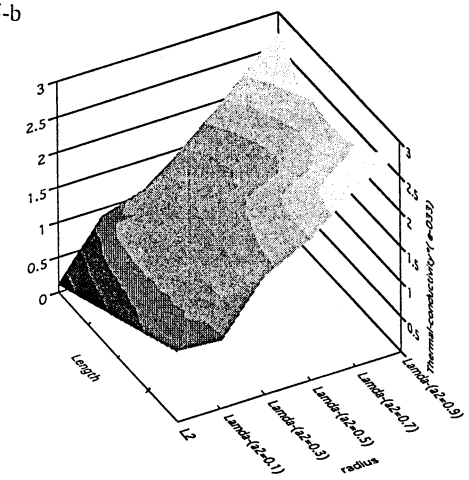


Fig 4. Normalized shear viscosity

Fig 4-a shows normalized shear viscosity $\eta/(5\,k\,T/2)$ as functions of mass (m2) and number density (n2) of the second particle. Fig 4-b shows those as functions of radius (a2) and length (L2) of the second particle. **Fig 5.Normalized thermal conductivity (timed by the factor 10^{-3})** Fig 5-a shows normalized thermal conductivity $4\lambda/(3\,k^3\,T^3)$ as functions of radius (a2) and mass distribution parameter ($\delta 2$) of the second particle. Fig 5-b shows those as functions of Length (L2 =0.1 to 0.9) of the second particle. The shear viscosity changes multi modally.

Fig 5-a shows normalized thermal conductivity $4\lambda/(3\,k^3\,T^3)$ as functions of radius (a2 =0.1 to 0.9) and mass distribution parameter ($\delta 2$ = 0.1 to 0.9) of the second particle. Fig 5-b shows those as functions of Length (L2 =0.1 to 0.9) of the second particle. The thermal conductivity has some minimum values in terms of the length of the second particle and changes complexly as a function of the radius of the second particle.

4. Discussion.

We have applied an established mathematical method [5],[7],[8] for evaluating the bio molecular interactions that relate to the gene expression, repressor, inducer particles. We have introduced minute explanations about the method particularly for the collision integral of the Boltzmann equation required for computing the shear viscosity and thermal conductivity. This is because any biological chemical interaction among the bio molecules starts from the encounter between the individual particle. Fluid dynamical approach such as creeping flow is too macro scopic to describe biological interactions at the molecular size. Electro magnetic and quantum approach are, on the other hand too micro scopic to describe an entire reaction system.

5. Conclusion.

A mathematical method was introduced for quantitative evaluation of gene expression in terms of thermo dynamical and statistical dynamical bio molecular interactions. By improving the present method, we can evaluate the human gene expression in quantitative manner.

6. References.

[1] Darnell J. (1990)

[2] Chapman S. and Cowling T G. (1952) The mathematical theory of non uniform gases. Cambridge .

[3] Curtiss C.F. (1956) Kinetic theory of non spherical molecules. J. Chem. Physics. 24. (2) : 225-241.

[4] Hirshfelder J.O. Curtiss C.F and Bird R B. (1954) Molecular theory of Gases and Liquids. Joh Wiley and Sons , Inc. N. Y.

[5] Muckenfuss C. and Curtiss C.F (1958) Kinetic theory of non spherical molecules III. J. Chem. Physics. 29. (6) : 1257-1272.

[6] Curtiss C.F and Muckenfuss C. (1957) Kinetic theory of non spherical molecules II. J. Chem. Physics. 26. (6) : 1619-1636.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Distributions of Electrical Reaction Potential, Circumferential Force and Free Energy of DNA Strands by Concentric Cylindrical Dielectric Continua Model.

H, Hirayama and *Y. Okita
Department of Public Health Asahikawa Medical College
*Shizuoka University

Abstract

We computed three dimensional distributions of electrical reaction potential, circumferential forces and free energy of DNA strand. The concentric cylindrical modeling of DNA was based on the continuum treatment of the ionic atmosphere of the DNA. Two Laplace equations in combination with one Poisson Boltzmann equations were solved by the analytic method. The potential and circumferential force oscillated as function of circumferential angle. The present method will be available for predicting the molecular interaction energy for the genetic expression.

key words : DNA, Helical strand, Potential, Free energy.

1. Introduction.

Genetic expression is regulated by the interaction of regulating bio molecular particle proteins and the DNA (Fig 1) strands. Both of them have definite charges around them and may determine the ultimate interaction process. Theoretical approaches for charge distributions around the DNA strands with cylindrical system have been one of the most attractive field not only in biology but also in physics. Particular importance has been emphasized on the spatial distribution of electrical charges on the surface of the DNA, the free energy of DNA and the circumferential force produced by the DNA when they rotates as a bio molecular motor to DNA replication.

In the present work, we apply the analytic method (Jarayam B 1990) and show the computed results of three dimensional distribution of electrical charge , circumferential force and free energy of DNA strand on the basis of a continuum treatment of the environment of the DNA.

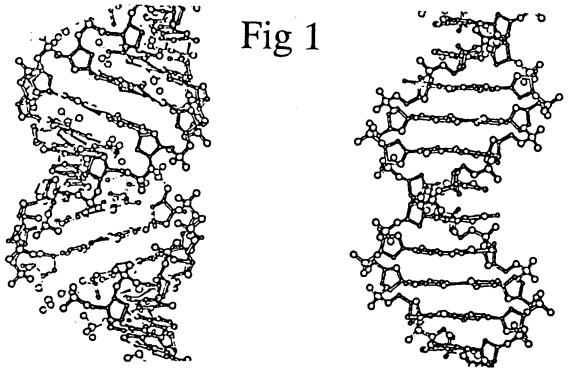


Fig 1

2. Modeling

The coaxial cylindrical system consisted of three regions. The inner region (Region A) describes a cylindrical cavity with L height, r =a radius and a dielectric constant εi where in the solute charges are located at sites rk. Region B is a local region extending radially from r =a to r=b. This region describes the first shell of the solute and contains solvent of dielectric constant ε loc. Region B constitute the ion exclusion zone. Region C is the outer region which ranges from r=b to r = infinite. This region contains solvent of dielectric constant εo and ion atmosphere including counter ion and co-ions.(Fig 2) The ion atmosphere enters into modeling through the Debye inverse length parameter κ. This is proportional to the square root of ionic strength. Now we solve the Laplace equation

$$\Delta \phi = 0 \text{ ---(1)}$$

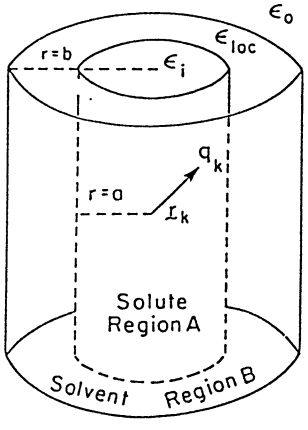
for regions A and B. We also have to solve the linearized Poisson-Boltzmann equation

$$(\Delta - \kappa^2)\phi = 0 \text{ ----(2)}$$

for region C. The general solutions for regions A and B are

$$\phi i \text{ and } \phi \text{loc} = \sum_{k=1} Bk / |r - rk| \text{ ---(3)}$$

Fig 2 Coordinates



Solvent and ion atmosphere
Region C

and for region C.

$$\phi o = \sum_{k=1} (A_k \exp(- \kappa | r - r_k |) / | r - r_k | + B_k \exp(- \kappa | r - r_k |) / | r - r_k |) \quad ---(4)$$

ϕi and ϕloc contain the Coulombic potential due to the discrete charge distribution. ϕ contains screened Debye potential. rk represents the Cartesian coordinates of the kth charge. Modifications to the Coulombics due to the presence of dielectric boundaries are introduced via the constants Ak and Bk. They are determined by adequate boundary conditions. The general solutions for the regions A, B and C in cylindrical coordinates are

$$\phi i = 1/\varepsilon_i \sum_{k=1} [-2q_k/L \log(\omega_k) + B_k + F_k + \sum_{n=1} \{ B_{nk} Io(\lambda \omega_k) + F_{nk} Io(\lambda \omega_k) + 4 q_k /L Ko(\lambda \omega_k) \} \cos(\lambda (z - z_k)) \quad ---(5)$$

$$\phi loc = 1/\varepsilon_{loc} \sum_{k=1} [-G_k \log(\omega_k) + F_k + \sum_{n=1} \{ F_{nk} Io(\lambda \omega_k) + G_{nk} Ko(\lambda \omega_k) \} \cos(\lambda (z - z_k)) \quad --(6)$$

and

$$\phi o = 1/\varepsilon_o \sum_{k=1} [C_k Ko(\lambda \omega_k) + \sum_{n=1} \{ C_{nk} Ko(\lambda' \omega_k) \} \cos(\lambda (z - z_k)) \quad ----(7)$$

where Ko(s) and Io(s) are the modified Bessel functions of zero order. The terms in Bk originate in the polarization of the local dielectric. Terms in Fk originate in the polarization of the bulk di electric continum. The cylindrical coordinates of the kth charge are described by ρ_k, ϕ_k and z_k. In addition

$$\omega_k = [\rho^2 + \rho_k^2 - 2 \rho \rho_k \cos(\phi - \phi_k)]^{1/2} \quad ----(8)$$

$$\lambda = 2\pi/L, \quad \lambda' = (\lambda^2 + \kappa^2)^2 \quad ----(9)$$

The constants B,F, G and C are determined by the boundary conditions. The reaction potential is

$$\phi R = 1/\varepsilon_i \sum_{k=1} [B_k + F_k + \sum_{n=1} \{ (B_{nk} + F_{nk}) Io(\lambda \omega_k) \} \cos(\lambda (z - z_k)) \quad ---(10)$$

The electric Helmholtz free energy of the charge distribution is

$$AeL = 1/2 \sum_{\nu=1} q_\nu \phi_R(r_\nu) \quad ---(11)$$

$\phi_R(r_\nu)$ is the reaction potential at the charged site r_ν of charge q_ν. The matching conditions for this problem are that the potential and dielectric displacement across the boundaries be continuous. This requires

$$(\phi i)_{\rho=a} = (\phi loc)_{\rho=a} \quad ---(12)$$

$$(\phi loc)_{\rho=b} = (\phi o)_{\rho=b} \quad ---(13)$$

$$\varepsilon_i (d\phi i/d\rho)_{\rho=a} = \varepsilon_{loc} (d\phi loc/d\rho)_{\rho=a} \quad ---(14)$$

$$\varepsilon_{loc} (d\phi loc/d\rho)_{\rho=b} = \varepsilon_{loc} (d\phi o/d\rho)_{\rho=b} \quad ---(15)$$

In applying these boundary conditions

$$\log(1/\omega_k) = \log(1/\rho_>) + \sum_{m=1} 1/m (\rho_< / \rho_>)^m \cos(m(\phi - \phi_k)) \quad ---(16)$$

$$Ko(\lambda \omega_k) = Io(\lambda \rho_<) + Ko(\lambda \rho_>) + 2 \sum_{m=1} \cos(m(\phi - \phi_k)) Im(\lambda \rho_<) Km(\lambda \rho_>) \quad --(17)$$

$$Io(\lambda \omega_k) = Io(\lambda \rho_<) + Io(\lambda \rho_>) + 2 \sum_{m=1} (-1)^m \cos(m(\phi - \phi_k)) Im(\lambda \rho_<) Im(\lambda \rho_>) \quad --(18)$$

Utilizing these expressions, the reaction potential can be

$$\phi R(r_k) = 1/\varepsilon_i \sum_{k=1} \Big[B_k + F_k + \sum_{m=1} \cos(m(\phi - \phi_k)) (B^m_k + F^m_k) \rho^m/m + \sum_{m=1} \{ (B_{nk} + F_{nk}) Io(\lambda \rho_k) Io(\lambda \rho) \cos(\lambda (z - z_k)) + 2 \sum_{m=1} \cos(m(\phi - \phi_k)) (-1)^m \sum_{n=1} (B^m_{nk} + F^m_{nk}) Im(\lambda \rho_k) Im(\lambda \rho) \cos(\lambda (z - z_k)) \Big] \quad ---(19)$$

Using the equations (5) to (8), (12) to (18), the coefficients in (19) are

$$B_k = 2q_k /(L \varepsilon_b) (1 - \varepsilon_a)/ \varepsilon_a \{ Ko(\kappa b) /(\kappa b K1(\kappa b)) + \varepsilon_b \log(b/a) \} \quad ---(20)$$

$$F_k = 2q_k /(L \varepsilon_b)\{ Ko(\kappa b) /(\kappa b K1(\kappa b)) + \varepsilon_b \log(b) \} \quad ---(21)$$

$$B^m_k = 2q_k /L (1 - \varepsilon_a)/ \varepsilon_a (\rho_k^m/a^{2m}) [1 + (a/b)^{2m} \{ (m Km(\kappa b) + \varepsilon_b \kappa b Km'(\kappa b))/(m Km(\kappa b) - \varepsilon_b \kappa b Km'(\kappa b)) \}] \quad ---(22)$$

$$F^m_k = 2q_k /L (\rho^m/b^{2m}) [\{ (m Km(\kappa b) + \varepsilon_b \kappa b Km'(\kappa b)) /(m Km(\kappa b) - \varepsilon_b \kappa b Km'(\kappa b)) \}] \quad --------(23)$$

$$B^m_{nk} = (-1)^m 4 q_k /L (\varepsilon_a' - 1) Km'(\lambda a) / Im'(\lambda a) \quad --(24)$$

$$F^m_{nk} = (-1)^m 4 q_k /L \varepsilon_a' \theta \quad ---(25)$$

$$B_{nk} = B^{m=0}_{nk} \quad \text{and} \quad F_{nk} = B^{m=0}_{nk} \quad ----(26)$$

$$\theta = [(\varepsilon_b \lambda' Km'(\lambda' b) Km(\lambda b) - \lambda Km(\lambda' b) Km'(\lambda b)) / (- \varepsilon_b \lambda' Km'(\lambda' b) Im(\lambda b) + \lambda Km(\lambda' b) Im'(\lambda b)) \quad ----(27)$$

$$\varepsilon'_a = \varepsilon_a /(\lambda a) 1/ [(1 - \varepsilon_a) \theta Im(\lambda a) Im'(\lambda a) + Km(\lambda a) Im'(\lambda a) - \varepsilon_a Km'(\lambda a) Im(\lambda a)] \quad ------(28)$$

$$\varepsilon_a = \varepsilon_{loc} / \varepsilon_i \quad \text{and} \quad \varepsilon_b = \varepsilon_o / \varepsilon_{loc} \quad --(29)$$

Substituting above constants in (19) and (11), the electro static free energy is obtained as

$$AeL = \sum_{L=1, k=1} \Big[q_L q_k /(L \varepsilon_b \varepsilon_i) \{ (1 - \varepsilon_a)/ \varepsilon_a \{ Ko(\kappa b) /(\kappa b K_1(\kappa b)) + \varepsilon_b \log(b/a) \} + (Ko(\kappa b) /(\kappa b K_1(\kappa b)) + \varepsilon_b \log(b)) \} + q_L q_k /(L \varepsilon_i) \sum_{m=1} \cos(m(\phi_L - \phi_k))/m \{ ((1 - \varepsilon_a)/ \varepsilon_a$$

$$[\,1+(a/b)^{2m}\{(m K_m(\kappa b)+\varepsilon_b \kappa b K_m'(\kappa b))$$

$$/(m K_m(\kappa b)-\varepsilon_b \kappa b K_m'(\kappa b))\}]$$

$$+1/b^{2m}\{(m K_m(\kappa b)+\varepsilon_b \kappa b K_m'(\kappa b))/(m K_m(\kappa b)$$

$$-\varepsilon_b \kappa b K_m'(\kappa b))\}\rho_k^m \rho_L^m/m$$

$$+2 q_L q_k/(L \varepsilon_i)\sum_{n=1}\{((1-\varepsilon''_a)K_1(\lambda a)/I_1(\lambda a)+\varepsilon''_a$$

$$\theta^{m=0,nk})I_0(\lambda \rho_k) I_0(\lambda \rho_L) \cos(\lambda(z_L-z_k))$$

$$+4q_L q_k/(L \varepsilon_i)\sum_{m=1}\cos(m(\phi_L-\phi_k))\sum_{n=1}((\varepsilon'_a-1)$$

$$K_m'(\lambda a)/I'_m(\lambda a)+\varepsilon'_a \theta) I_m(\lambda \rho_k) I_m(\lambda \rho_L)$$

$$\cos(\lambda(z_L-z_k))] \qquad \text{---(30)}$$

where

$$\theta^{m=0,nk}=[(-\varepsilon_b \lambda' K_1(\lambda' b) K_0(\lambda b)+\lambda K_0(\lambda' b)$$

$$K_1(\lambda b))/(\varepsilon_b \lambda' K_1(\lambda' b) I_0(\lambda b)$$

$$+\lambda K_0(\lambda' b) I_1(\lambda b)) \qquad \text{----(31)}$$

$$\varepsilon''_a=\varepsilon_a/(\lambda a)\,1/[(1-\varepsilon_a)\theta^{m=0,nk} I_0(\lambda a) I_1(\lambda a)$$

$$+K_0(\lambda a) I_1(\lambda a)+\varepsilon_a K_1(\lambda a) I_1(\lambda a)]$$

$$\text{-----(32)}$$

Putting

$$\alpha=(m K_m(\kappa b)+\varepsilon_b \kappa b K_m'(\kappa b))$$

$$/(m K_m(\kappa b)-\varepsilon_b \kappa b K_m'(\kappa b)) \qquad \text{-------(33)}$$

The potential in the inner region is

$$\phi_i=\sum_{k=1} 2 q_k/(L \varepsilon_i)[\,\mathrm{Log}(1/\rho)+\varepsilon_i/\varepsilon_o\{K_o(\kappa b)/(\kappa b$$

$$K1(\kappa b))\}+\varepsilon_i/\varepsilon_{loc}\log(b/a)+\log(a)\}$$

$$+\sum_{m=1}\cos(m(\phi-\phi_k))/m\{1+((1-\varepsilon_a)/\varepsilon_a(\rho/a)^{2m}$$

$$\{1+(a/b)^{2m}\alpha\}+(\rho/a)^{2m}\alpha)\}\rho_k^m/\rho^m$$

$$+2\sum_{n=1}\{((1-\varepsilon''_a)K_1(\lambda a)/I_1(\lambda a)+\varepsilon''_a \theta^{m=0,nk})$$

$$I_o(\lambda \rho_k) I_o(\lambda \rho)+I_o(\lambda \rho_k) K_o(\lambda \rho)]\cos(\lambda(z-z_k))$$

$$+4\sum_{m=1}\cos(m(\phi-\phi_k))\sum_{n=1}[((\varepsilon'_a-1)K_m'(\lambda a)$$

$$/I'_m(\lambda a)+\varepsilon'_a \theta) I_m(\lambda \rho_k) I_m(\lambda \rho)+I_m(\lambda \rho_k)$$

$$K_m(\lambda \rho)]\cos(\lambda(z-z_k))] \qquad \text{---(34)}$$

For computation, we used the reported measured parameters a = 5 A, b = 12 A, L =50 A, ε_i =2, ε_{loc} =20, ε_o =80, κ =8 A. A : angstrom (10-8.)

For molecular electro static motor, we obtained the force operating along the circumferential direction by differentiating the reaction potential with respect to θ. This force may dissociate the DNA double strand when the DNA template has been utilized on the DNA transcription.

3. Results

Fig 3 show the three dimensional representations of Free electrostatic energy (lkcal/mol Phosphate) of DNA as functions of circumferential angle ψ rad), axial distance (zL Angstrom) and radial distances (ρ : RohL Angstrom

Fig 3 Free energy

Fig 3-a

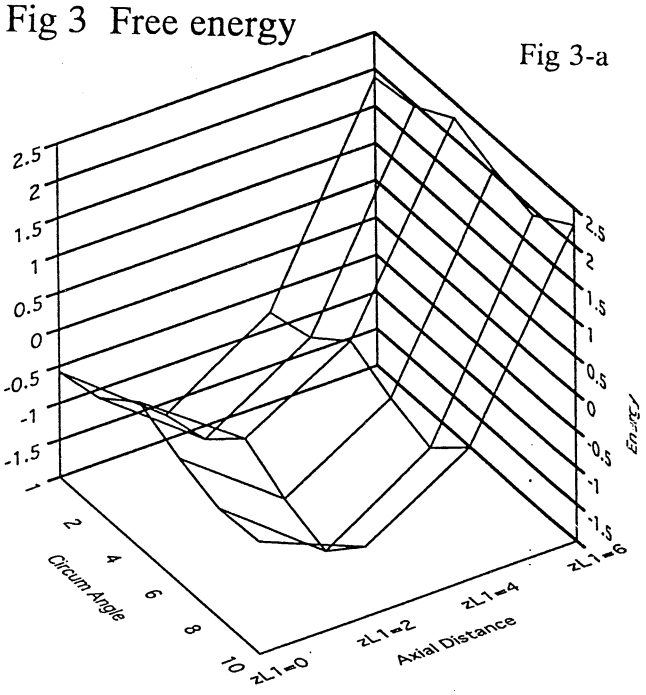


Fig 3-b

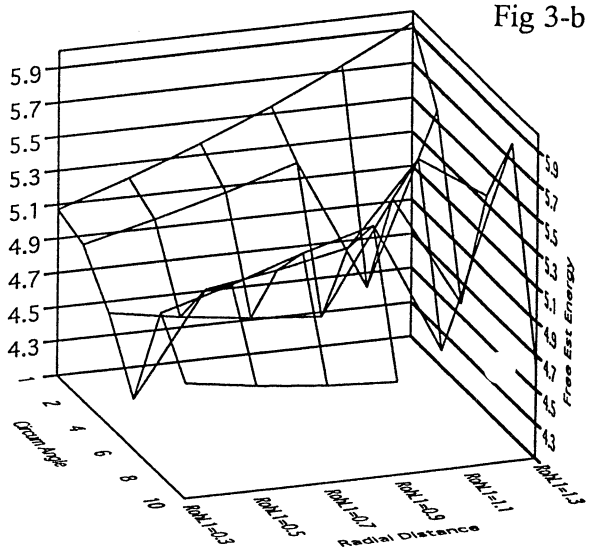


Fig 3-c

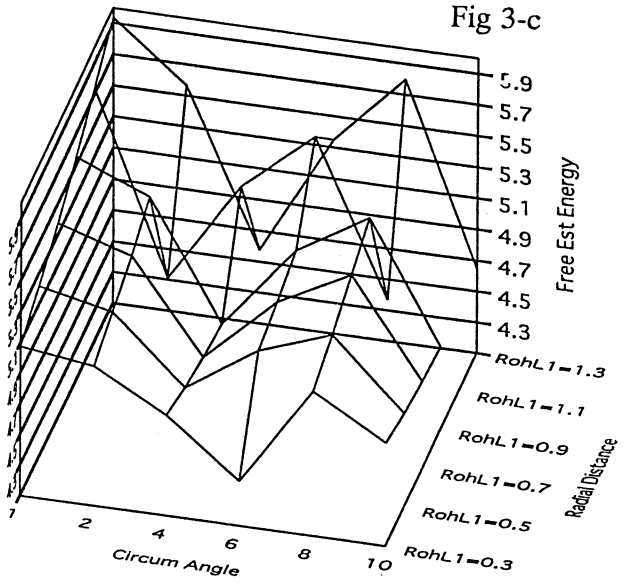


Fig 4 Potential

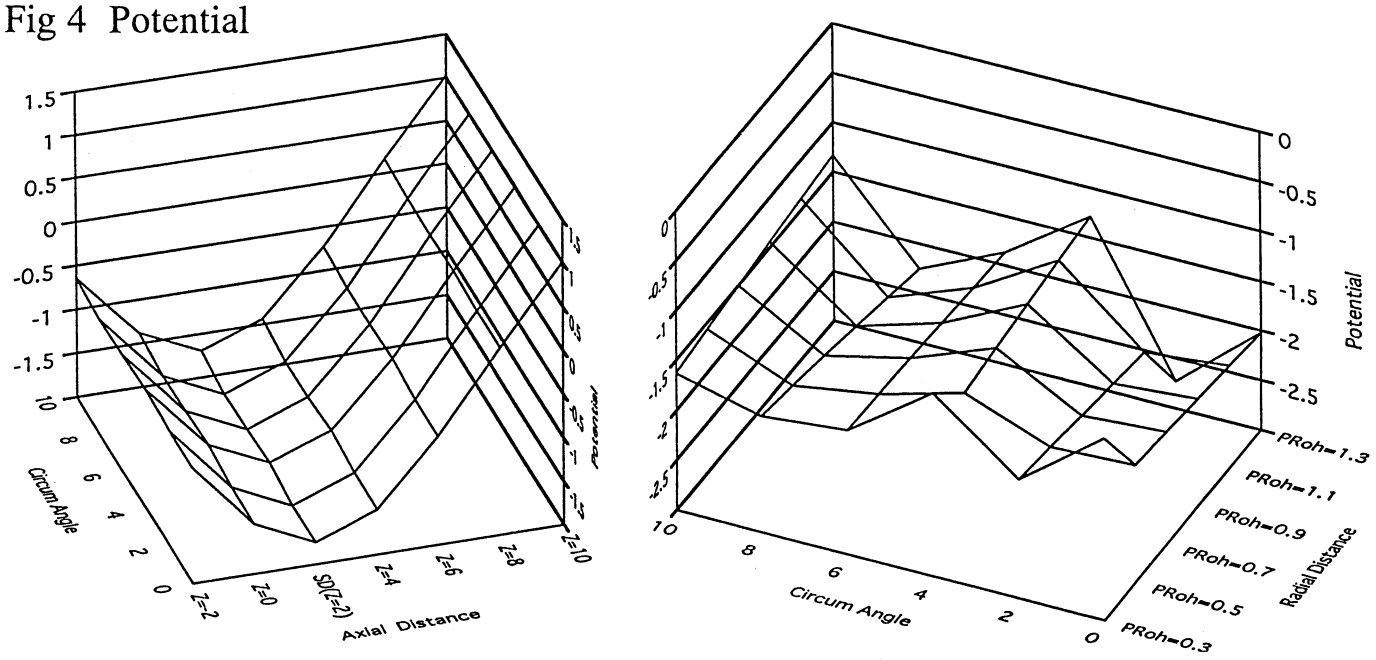
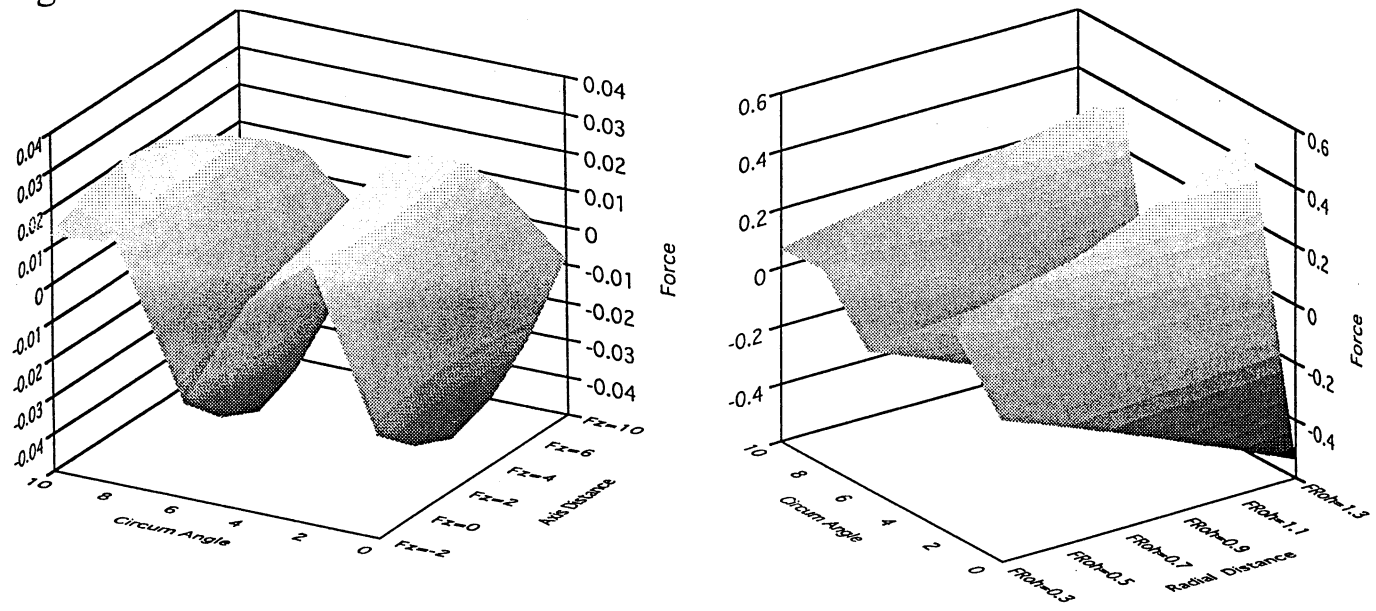


Fig 5 Circumferential Force



)per unit lengin of DNA per εb εi . The energy oscillated as function of circumferential angle. As the distance between reference point ρ =1.5 has shortened, the free energy increased (Fig 3-a , Fig 3-b) rapidly due to the Coulombic effect.

Fig 4 shows the reaction potential ϕ R(rk) normalized by εi as functions of circumferential angle, axial distance (Fig 4-a) and radial distance (Fig 4-b). AS in the free energy, the reaction potential oscillated in the circumferential direction. Significant elevation appeared when the axial distance from the reference point (z =1.5) has shorted.

Fig 5 shows the three dimensional changes in circumferential force as functions of circumferential angle, axial distance and radial distance. The circumferential force oscillated regularly as functions of circumferential angles. Particularly, the alternative changes between negative and positive values of the force may contribute to the repulsive and attractive interaction between the mutual facing bases that have charges in opposite signs.

5. Conclusion.

The free energy, reaction potential and circumferential force of DNA strand oscillate as function of circumferential angle and changes rapidly as the axial and radial distance increase.

6. References

1. Jarayam, B and Beveridge, DL. J. Phys. Chem. vol 94. pp 4666-4671. 1990.

[APPENDIX]

The recurrent formula for modified Bessel functions

$K_m(z)' = (m/z) K_m(z) - K_{m+1}(z)$

$I_m(z)' = (m/z) I_m(z) + I_{m+1}(z)$

$K_{m+1}(z) = 2\ m/z\ K_m(z) + K_{m-1}(z)$

$I_{m+1}(z) = -2\ m/z\ I_m(z) + I_{m-1}(z)$

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

USING TDGL FOR ESTIMATION OF BLOOD HEAT DIFFUSION

[1]Zamri Zainon and [2]Mohamed Rizon
[1]Department of Mechanical Engineering, University of Malaya, 50603 Kuala Lumpur, Malaysia
Tel : 603-7959 5281 Email : zamri@linuxfan.com
[2]Department of Biomedical Engineering, University of Malaya, 50603 Kuala Lumpur, Malaysia
Tel : 603-7959 4580 Email : mohamed@fk.um.edu.my

Abstracts
An attempt has been made to simulate the heat diffusion through a small tube such as blood vessel and it was conducted with the practical importance where the tube was set cylindrically. With application of the statistical method that predicts the situation and randomly chosen condition, the heat at the area was calculated based on the free energy of the blood flow. The results revealed that the heat diffuse scatter accordingly with the radial condition and changes with time.

Keywords: statistical mechanics, approximation methods, computational models, heat diffusion, dynamics of blood artificial organs.

I. Introduction

For a cardiovascular physician, they may required to study the workload in the blood tube such as transportation of internal heat generated from human activities and metabolism to various part of the body. On the other hand, for an engineer that works on developing the artificial organs etc; particularly in the field of thermofluid dynamics, the interest of this study is to focus on how the generated heat distributed through the cross sectional area of the vessel. The final target of this study is to the estimate the authorize blood flow rate in the vessel particularly for the artificial organ with suit individual physical condition. Among the methods in statistical mechanics, we implied the Time Dependent Ginsburg-Landau (TDGL) for this purpose since it has been a popular tool for state condition estimation and energy distribution. Besides, this method might be suitable to estimate the complex situation in a micro scale tube such as blood vessel.

There are also several works on dynamics of blood flow, experimentally or simulation via electronics computation. Most of the studies are based on the animal testing for example, Brook et al. [1] that study the blood dynamics in a long giraffe's neck. The electrical characteristics in the blood was studied by Sakamoto et al. [2] with approach on effects of erythrocyte orientation and deformation an axial accumulation that cause the differences in resistance between the flowing and stagnant blood. They concluded that the orientation effects on impedance pletsysmography pulse waveforms nearly equal that of the blood vessel ammeter change and that waveform analysis must consider both effects. The measurement on velocity profiles of the flowing blood by Angleson et al. [3] was conducted using the Doppler frequency estimations in ultrasonic moving target indicator measurement of velocity profiles in the flowing blood. While Petrofsky [4] successfully measured the regional blood flows in the brain of a cat, based on the principle of the photoelectric pletsysmograghy which is comparatively a less brutal method than others such as injection of radiative tags [5] and radiative micro pulse [6] that offer an excellent estimation but require the sacrifice of the animals.

On the other hands, as our purpose for the applications of the artificial organs, particularly for wide range of the human bodies conditions, we propose a computational techniques that are safer and unhurt like the above mentioned techniques.

II. Constitutive Equations

As an incompressible fluid, the dynamics of blood flow should follow the conservation of mass as shown by,

$$\frac{\partial A}{\partial t}+\frac{\partial uA}{\partial x}=0 \qquad (1)$$

where A is the cross-sectional area of the tube, $u=u(x,t)$ is the velocity of the blood. Conservation of momentum can be shown as,

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+\frac{1}{\rho}\frac{\partial p}{\partial x}+\frac{1}{\rho}R(A,u)uA-g=0 \qquad (2)$$

where p is the internal pressure, ρ is the density of blood and g is the gravitational acceleration. The term, $R(A,u)>0$ represent the viscous resistance to the flow per unit length of the tube. The resistance is termed linearly by Brook et al. [1] as

$$R=\frac{8\pi u}{A^2} \qquad (3)$$

Let us consider the laws of thermodynamics connecting a specific internal energy E, specific entropy S, absolute temperature T, pressure p, specific volume V, density ρ, stress σ_{ij} and strain e_{ij} and refer to Fig. 1 to examine one steady state of flow condition.

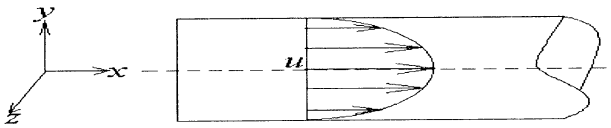


Fig. 1 : Velocity profile in a steady laminar flow into cylindrical tube.

The first law of thermodynamics – the law of conservation energy states that in the given body of material of unit volume, an infinitesimal change in the internal energy is equal to the sum of heat transferred to the body, dQ and work done by the body dW, which is equal to the product of stress σ_{ij} and the change of strain de_{ij}. Fung [7] expressed all the quantities above in unit mass as,

$$dE = dQ + \frac{1}{\rho}\sigma_{ij}de_{ij} \tag{4}$$

While the second law of thermodynamics states that the heat-input dQ is equal to the product of the absolute temperature and the change of entropy,

$$dQ = TdS \tag{5}$$

Combining the two expressions, we get

$$dE = TdS + \frac{1}{\rho}\sigma_{ij}de_{ij} \tag{6}$$

In the isothermal condition, the temperature and strain are considered as the independent variables; hence we can transform Eq. (5) as the specific free energy;

$$F = E - TS \tag{7}$$

and get,

$$dF = dE - TdS - SdT \tag{8}$$

or in the form of heat equation from Eq. (5) as

$$dQ = dE - dF - SdT \tag{9}$$

III. Statistical Methods

The TDGL method [8] applied in this study deals with the estimation of state condition and hence it is necessary to assume the partition function with the average value. For the internal energy it can be defined as,

$$< E >= \frac{\sum EW(E)e^{-E/kT}}{\sum W(E)e^{-E/kT}}$$

$$= kT^2\frac{\partial}{\partial T}\sum W(E)e^{-E/kT} \tag{10}$$

While the heat capacity can be expressed as the variation function as,

$$\frac{d < E >}{dT} = \frac{1}{kT^2}\left(< E^2 > - < E >^2\right) \tag{11}$$

In TDGL method, the free energy derived from the heat capacity is then determined for each state, by the power function as follows,

$$F(s) = F_0 + \nabla s + Bs^2 + Cs^3 + Ds^4 + ... \tag{12}$$

where F_0 is the initial free energy, B, C, D are constants and ∇ denotes the Laplacian. The parameter of condition s, which shows zero in value, is referred as having a high temperature and in a chaotic behavior. On the other hands for the $0 < s \le 1$ and $-1 \le s < 0$ in value shows the lower temperature and in good order of behavior.

To apply the heat diffusion, the temperature difference ($\tau = T_c - T$) with T_c as the flowing blood temperature, is considered to be very essential where this value affects the phase transition.

IV. Calculations

The calculation is carried out for the practical importance in biomechanics with a simple condition of cylinder tube and thus assumed that the blood molecule agree with the thermodynamic law as formulated in Eq. (1) ~ (9). As the heat diffuses through the cross sectional area of the blood vessel, the free energy is then considered as the energy with the relation to the state and spatial conditions, which can be expressed as,

$$F(s) = \iiint\left\{\tau\left[s(\vec{r})\right]^2 + D\left[s(\vec{r})\right]^4 + \sigma_{ij}\left[\nabla s\right]^2\right\}d\vec{r} \tag{13}$$

and we get,

$$\frac{dF}{ds} = -2\tau s - s^3 + 2\sigma_{ij}\nabla s \tag{14}$$

On the other hand, the TDGL, which is basically the method based on the situation changes with time, is expressed as,

$$s(x,t + \Delta t) = s(x,t) + ds(x,t)\Delta t \tag{15}$$

and, the differential against time t as,

$$\frac{ds}{dt} = \tau s - s^3 + \sigma_{ij}\nabla s \tag{16}$$

The calculation as discussed above is for non-conservative Time Dependent Ginnsburg-Landau.

For a steady state with a conservation of momentum and condition including the parameter of the calculation, the following procedure is suggested for state equation,

$$\frac{ds}{dt} = \nabla s\left(\tau s - s^3 + \sigma_{ij}\nabla s\right) \qquad (17)$$

where ∇s is the Laplacian given as,

$$\nabla s = \frac{s(r+dr) + s(r-dr) - 2s(r)}{(dr)^2} \qquad (18)$$

For estimation of the heat diffusion, the initial blood temperature is assumed to be constant, and for every τ ($=T_c$-T) the heat is diffused through the cross sectional area of the tube.

We then tested the result on the Spatial Correlation Function for the both TDGL at $\tau = 1.0$ to examine the characterized length (distance), $L(t)$ of the spatial condition. The spatial correlation function, G for these cases can be written as,

$$G(\vec{r},t) = < S(\vec{r_0},t)S(\vec{r_0}+\vec{r},t) >_{r_0} \qquad (19)$$

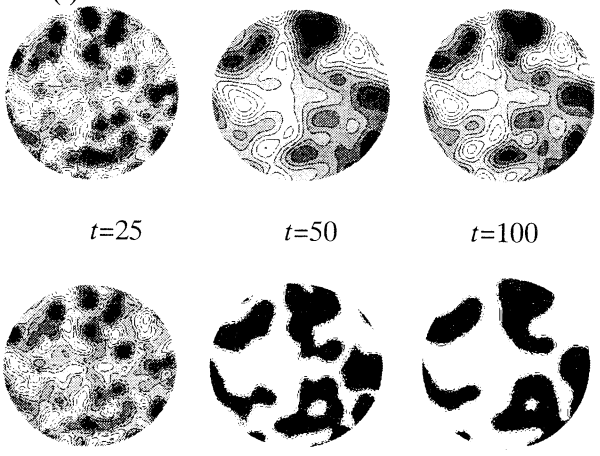
where the mean value for G is evaluated at r_0.

V. Results and Discussion

All the numerical data were then graphically visualized with the heat energies evaluated based on the state condition.

(a) Non-Conservative

(i) τ = 1.0



t=25 t=50 t=100

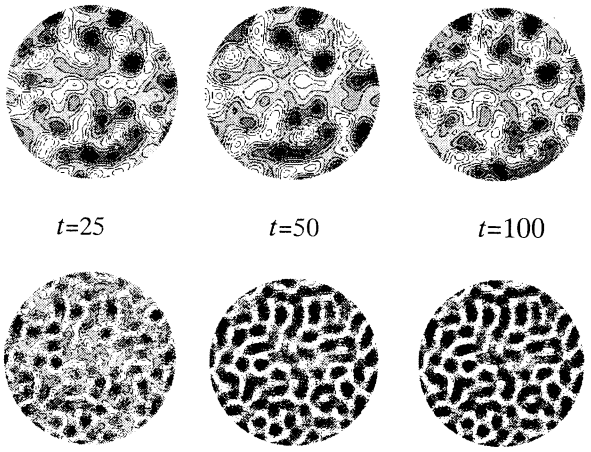
(ii) τ = -1.0

Fig. 2: Visualization of heat diffusion corresponding to time for non-conservative TDGL for (i) τ =1.0 and (ii) τ =-1.0

The simulation was set for a very small tube with radius of 1μm and the time dependent was set for 100 steps of the calculation. In these figures, the brighter spot indicates higher heat energy and the darker spot indicates the lower heat energy respectively. The result for non-conservative TDGL revealed that the heat energy scatter considerably corresponding to the radius of the tube. For τ =1.0, it seems that the heat was not diffused to reach the "instability" as shown in Fig. 2 for τ =-1.0 at the t=100, where the heat spread widely with two clearly different value. However the transition can also be seen but it was more slowly to compare with how Fig. 2(ii) shows.

(b) Conservatives

(i) τ =1.0



t=25 t=50 t=100

(ii) τ =-1.0

Fig. 3: Visualization of heat diffusion corresponding to time for conservative TDGL for (i) τ =1.0 and (ii) τ =-1.0

As the parameters of situation for the calculation was set conservatively it may agree with the law of momentum conservation as in Eq. (2) and hence the figures show that there is no drastic changes in the heat diffusion for in these cases. The heat are diffused accordingly with considerable order in the cross sectional area of the tube and changes with time for the both of the given temperature differences. Each of the situations also has different style of figures, which reveals that the heat diffusion for all the situation are not the same and indicates that they are in different size.

The results on the spatial correlation function of the TDGL at $\tau = 1.0$, are shown in Figs. 4. For the conservative TDGL we can see that G reaches the negative value where the $\pm$ value in this situation is due the calculated state condition. This indicates that the phase transition occurs but with in the same

temperature condition as described in section III. For the non-conservative TDGL, it does not clearly show the same effect, and hence we can conclude that the only +ve values of the state condition exist. However for a bigger τ in conservative TDGL, we might observe a different result since the transition space is less wide. While for non-conservative TDGL, this phenomenon would not show out since the transition space is becoming wider as the time is consumed as shown in Fig. 2.

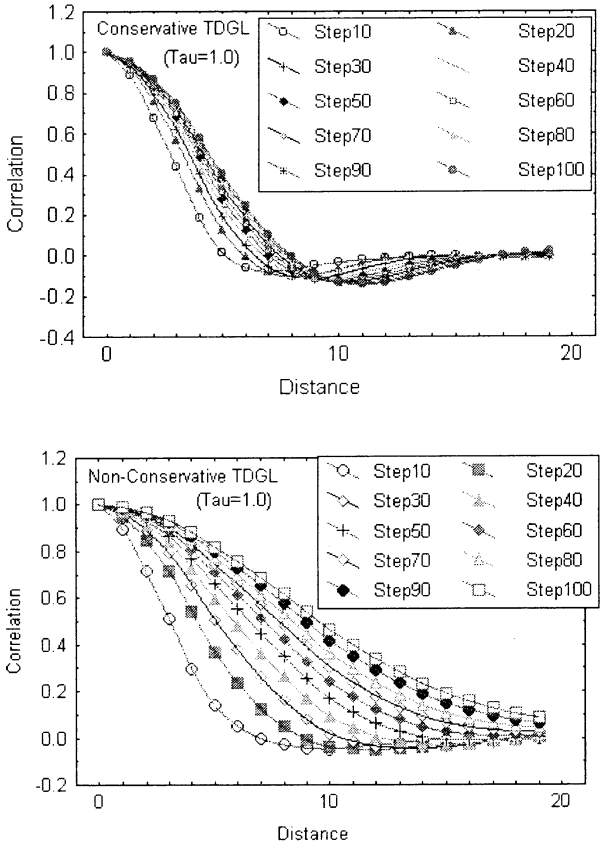


Fig 4 : Spatial Correlation Function for the both TDGL

VI. Conclusion

The estimation of the heat diffusion in the cross sectional area of the blood tube is not so simple. It would be very difficult to calculate through the direct numerical simulation since the treated area was very small and hence the statistical method was found to be more suitable. Through the calculation it is found that the heat diffuse scatter with the radial condition and changes with the time. The method introduced in this study is a rough estimation without consideration of turbulence phenomena or any organic effects in the blood flow. Moreover, the blood vessel configuration and flow instabilities estimation condition are the other assignment left for a further research based on this idea.

References

[1]Brook, B.S., Falle, S.A.E.G., and Pedley, T.J., Unsteady gravity-driven flows in collapsible tubes, J. Fluid Mech. 1999, vol.396, pp223-256.

[2]Sakamoto, K. and Kanai, H., Electrical Characteristics of the flowing blood, IEEE Transaction on Bio-Medical Eng., 1979, vol. BME-26, no. 12, pp. 686 – 695.

[3]Angelson, B. and Kristoffen, K., On ultrasonic measurement of velocity profile in blood flow, IEEE Transaction on Bio-Medical Eng., 1979, vol. BME-26, no. 12, pp. 665 –671.

[4]Petrofsky, J. S., In vivo measurement of brain blood flow in the cat, IEEE Transaction on Bio-Medical Eng., 1979, vol. BME-26, no. 12, pp. 441 – 445.

[5]Elkof, B., Lassend N.A., Nilson, L., Norburg, K., Srepzo, B.K., Torlof, P., Cerebral blood flow in the rat measured by tissue sampling techniques, 1974, Acta Physiol. Scand, vol. 91, pp. 1 –81.

[6]Alan, A. and Bill, A., Effects of Stimulation of the cervical sympathetic chain on retinal oxygen tension and on uveal cerebral blood flow in cats, 1973, Acta Physiol. Scand, vol. 91, pp. 84-91.

[7]Fung, Y.C., Biomechanics - Mechanical Properties of Living Tissues, Springer-Verlag, 1981, pp214-216.

[8]Matsumoto, M., Statistical Thermodynamics Lecture Notes, 1999, Graduate School of Engineering, Kyoto University.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Equation for Resting Membrane Potential of Cell

X. Zhang and H. Wakamatsu

Faculty of Medicine, Tokyo Medical and Dental University
1-5-45 Yushima Bunkyo-ku, Tokyo 113, JAPAN
E-mail zhang.mtec@med.tmd.ac.jp

Abstract

A new generalized equivalent circuit of cell, different from the Hodgkin-Huxley's one, is proposed. Using this equivalent circuit, we derive a general equation to relate resting membrane potential with capacity of ion-pumps, intracellular impermeant anions concentrations, extracellular ions concentrations, and permeability of ions. Influence of extracellular ions concentrations on the resting membrane potential was calculated using the equation. The results questioned the validity of using GHK (Goldman-Hodgkin-Katz) equation for bio-membranes.

1. Introduction

Any living cell has a resting potential across its membrane due to the existence of ion-pumps and intracellular impermeant anions. However, a relation between resting membrane potential with ion-pumps and impermeant anions has not been discussed yet. In this paper, we propose a generalized equation to calculate resting membrane potential by relating intracellular impermeant anions concentrations, extracellular ions concentrations, membrane ions permeability, and active ions currents generated by ion-pumps in the steady state. To make the concept clear, resting membrane potential discussed in this paper is the potential of cell in the steady state.

Resting membrane potential of cell has been thought to be estimated accurately using GHK (Goldman-Hodgkin-Katz) equation (Hodgkin et al., 1959; Hall, 1992). However, in this paper a misinterpretation will be elucidated, since GHK equation will show not to be suitable for bio-membranes.

2. Concept of *ion-condenser*

In Hodgkin-Huxley's equivalent circuit, voltage sources have been used to represent equilibrium potential of ions. However, as long as a cell continues being excited, ions concentrations gradients become lower, namely, the equilibrium potentials of ions decrease. What recovers the ions concentrations gradients are ion-pumps (Hall, 1992; Alberts et al., 1994). In order to find how the resting potential relates with ion-pumps and intracellular impermeant anions, permeable ions concentrations gradients, namely, the equilibrium potential of ions, have to be defined as variables. In this case, using constant voltage sources to represent the equilibrium potential of ions, as in Hodgkin-Huxley's equivalent circuit, is not suitable. Assuming that extracellular ions concentrations are constant, the concentration gradient of an ion across a cell's membrane, namely, the equilibrium potential, depends on the quantity of the ion in the cell and volume of the internal fluid. This characteristics appears in a condenser, in which voltage depends on the quantity of electrical charge and capacity. Thus, in this paper, we introduce a type of condenser to represent ions concentration gradients, and its voltage to represent the chemical potential (opposite to equilibrium potential). We named this condenser as an ***ion-condenser***.

The chemical potential V_X of an ion X is obtained from Nernst relation

$$V_X = \frac{RT}{z_X F} \ln \frac{[\mathrm{X}]_i}{[\mathrm{X}]_o} \tag{1}$$

where $[\mathrm{X}]_i$ and $[\mathrm{X}]_o$ are intra and extracellular concentrations of X; R, T and F are gas constant, absolute temperature and Faraday constant, and z_X is valence of X.

The intracellular electrical charge Q_X of ion X is

$$Q_X = z_X F v([\mathrm{X}]_i - [\mathrm{X}]_o) \tag{2}$$

where v is volume of the internal fluid of the cell.

From Equations (1) and (2), we get the relation between Q_X and V_X

$$Q_X = z_X F v [X]_o \left(e^{\frac{z_X F}{RT} V_X} - 1 \right) \tag{3}$$

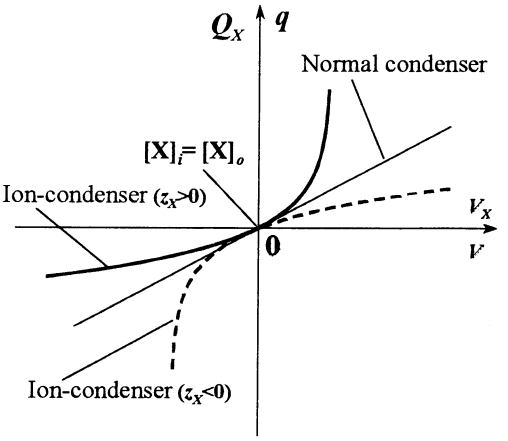


Figure 1. Characteristics of an *ion-condenser*.

Equation (3) shows that the characteristics of an *ion-condenser* are different from the normal condenser ($q=CV$), i.e., an *ion-condenser* is a nonlinear condenser. The characteristics of an *ion-condenser*, as shown in Figure 1, depend on the extracellular concentration of the ion, volume of the internal fluid of the cell, temperature, and ion valence.

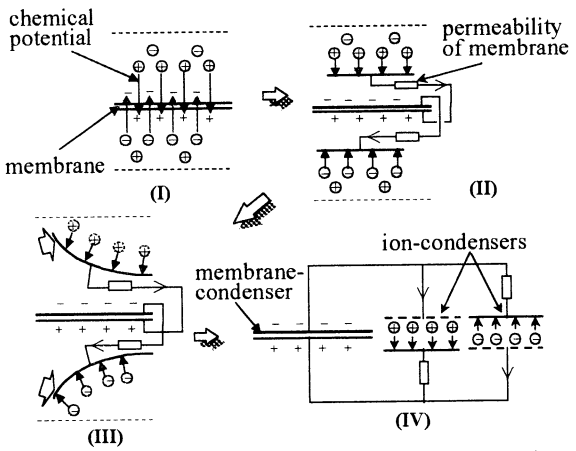


Figure 2. Relation between *ion-condenser* and membrane condenser.

Figure 1 shows that voltage of an *ion-condenser* approaches to a constant value when $[X]_i >> [X]_o$. On the other hand, if $[X]_i << [X]_o$, voltage of an *ion-condenser* becomes very sensitive to variation of Q_x, i.e., to ion X current I_X. This means that when an ion's intracellular concentration is much lower than its extracellular concentration, say, Na^+ or Cl^-, a very small quantity of the ion's flow through the cell's membrane will cause a large variation of the chemical potential. Thus, **only when ion's intracellular concentration is much higher than its extracellular one, the *ion-condenser* can be considered as a voltage source.** Figure 2 shows how we synthesized an electrical equivalent circuit of a cell membrane. Figure 2(I) shows fluids in different concentrations of two types of ions separated by a membrane. Electrochemical potential causes the ions to diffuse through the membrane. As in Hodgkin-Huxley's equivalent circuit (Hodgkin, 1962; Hodgkin et al., 1952; Hodgkin et al., 1952), we represented electrical capacity of the membrane as a condenser, and permeability of the membrane, as conductance, so circuits were drawn to have equal permeability through membrane as shows Figure 2(II). If we represent chemical potential of the ions as the electrical potential of an *ion-condenser*, Figure 2(II) can be drawn as Figure 2(IV).

3. Equivalent circuit of cell

Figure 3(A) shows the flux of ions in a cuttlefish giant neuron (Zhang et al., 1999; Hall, 1992; Alberts et al., 1994). We assume that the ion-pumps in the cell are only Na^+-K^+-pumps, and the extracellular permeable ions are Na^+, K^+, Ca^{++}, Mg^{++}, and Cl^-. Y^- represents the summation of intracellular impermeant anions. We consider the Na^+-K^+-pumps as two-current sources.

From the viewpoint of electrical engineering, current sources of ion-pumps are connected in parallel with ion-channels, and in series with *ion-condenser*s. From Figure 2(IV), we draw the equivalent circuit of Figure 3(A) as Figure 3(B), where C_{Na}, C_K, C_{Ca}, C_{Mg}, and C_{Cl} represent *ion-condenser*s of Na^+, K^+, Ca^{++}, Mg^{++}, and Cl^-, equivalent conductance g_{Na}, g_K, g_{Ca}, g_{Mg}, and g_{Cl} represent permeability through membrane of Na^+, K^+,

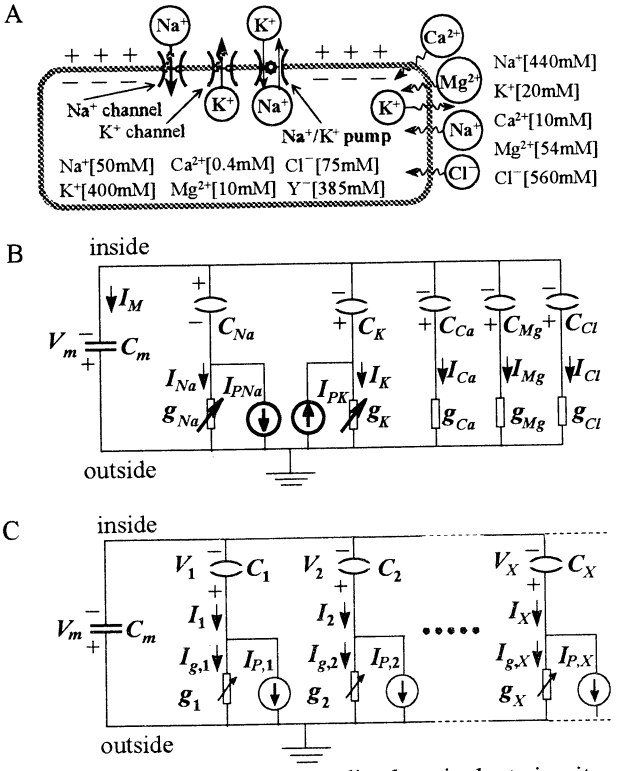


Figure 3. Structure of a generalized equivalent circuit based on electrical characteristics of cell.

Ca^{++}, Mg^{++}, and Cl^-, and C_m represents the membrane electrical capacity. $I_{P,Na}$ and $I_{P,K}$ are currents of Na^+ and K^+ actively transferred by Na^+-K^+-pumps.

From Figure 3(B), we obtain a general equivalent circuit to represent any type of cells as shows Figure 3(C), where X (X =1, 2, ···) represents permeable ions in the external fluid, $I_{P,X}$ represents active transport current, and $I_{g,X}$ represents passive transport current of ion X. If X has no active transport, $I_{P,X}$=0, and if ion X has no ion channel, conductance of ion X through membrane g_X is constant.

4. Equation of resting membrane potential

Since impermeant anions do not influence the structure of an equivalent circuit (Zhang et al., 1999), in this section, we calculate the resting membrane potential using the equivalent circuit of Figure 3(C).

If we represent the electrical charge of an impermeant anion Y as Q_Y, the following equations are written

$$\sum Q_X + \sum Q_Y = q_m \tag{4}$$

$$Q_Y = vFz_Y\left([Y]_i - [Y]_o\right) \tag{5}$$

where g_m is electrical charge on membrane, and z_Y is valence of Y. From Equations (3) ~ (5), the following equation is obtained:

$$\sum_X z_X[X]_o\left(e^{\frac{z_X F}{RT}V_X} - 1\right) + \sum_Y z_Y\left([Y]_i - [Y]_o\right) = \frac{q_m}{vF} \tag{6}$$

Since q_m is far smaller than ΣQ_X (Hall, 1992), and the external fluid can be thought as neutral, i.e.,

$$\sum_X z_X[X]_o + \sum_Y z_Y[Y]_o = 0 \tag{7}$$

Equation (7) becomes

$$\sum_X z_X [X]_o e^{\frac{z_X F}{RT} V_X} + \sum_Y z_Y [Y]_i = 0 \quad (8)$$

From the equivalent circuit of Figure 3(C), we obtain the following equations

$$I_{g,X} + I_{P,X} = I_X \quad (9)$$

$$V_X = -V_m + \frac{I_{g,X}}{g_X} \quad (10)$$

In the steady state, the concentrations of intracellular ions are constant, i.e., $I_X = 0$. Thus, from Equation (9) we obtain

$$I_{g,X} = -I_{P,X} \quad (11)$$

Hence, Equation (10) becomes

$$V_X = -\left(V_m + \frac{I_{P,X}}{g_X}\right) \quad (12)$$

From Equations (8) and (12), the generalized equation is obtained as

$$\sum\left(z_X [X]_o e^{-\frac{z_X F}{RT}\left(V_m + \frac{I_{P,X}}{g_X}\right)}\right) + \sum z_Y [Y]_i = 0 \quad (13)$$

Using Equation (13), we can calculate resting membrane potential V_m of any type of cell. Equation (13) shows that the resting membrane potential is determined by: 1) concentrations of extracellular ions and their valences, 2) concentrations of intracellular impermeant anions and their valences, 3) ratios of active transport current and permeability of each ion, and 4) temperature around the cell.

We can consider $I_{P,X}$ and g_X as either total active current and total conductance through the cell's membrane, or as active current and conductance per unit of area through the cell's membrane. Therefore, **the size of a cell has no influence on the resting membrane potential.** Calling the ratio of $I_{P,X}$ and g_X as active transport potential of ion X, and representing it as $V_{AT,X}$, Equation (13) can be expressed as

$$\sum\left(z_X [X]_o e^{-\frac{z_X F}{RT}(V_m + V_{AT,X})}\right) + \sum z_Y [Y]_i = 0 \quad (14)$$

5. Effect of extracellular ions concentrations on membrane potential

In the steady state, the membrane potential of a cell is constant, thus, the total electrical current through its membrane is 0, i.e.,

$$\sum I_X = 0 \quad (15)$$

This is the basal condition of GHK (Goldman-Hodgkin-Katz) equation. However, GHK equation has not considered currents of active transportation of ions, i.e., $\Sigma I_{g,X}$ was assumed as 0. From Figure 3(C) and Equation (15), the following equation can be obtained:

$$\sum\left(I_{g,X} + I_{P,X}\right) = 0 \quad (16)$$

Equation (11) shows that, in the steady state, the value of $I_{P,X}$ is equal to $-I_{g,X}$, and there is no reason to think $\Sigma I_{P,X}$ as 0, i.e., $\Sigma I_{g,X} = -\Sigma I_{P,X} \neq 0$. Thus it can be thought that GHK equation is not suitable for living bio-membranes.

Untill now, the resting membrane potential of a cell had to be estimated using the GHK equation as the following (Hodgkin, et al, 1959; Hall, 1992) :

$$V_m = \frac{RT}{F} \ln \frac{[K]_o + 0.01[Na]_o}{[K]_i} \quad (17)$$

or by omitting the $0.01[Na]_o$ as

$$V_m = \frac{RT}{F} \ln \frac{[K]_o}{[K]_i} \quad (18)$$

Curves for Equations (17) and (18) drawn in Figure 4 when using chloride-free sulphate solution containing 8 mM-$CaSO_4$ as described in Table 1, and the results of the experiments made by Hodgkin and Horiwicz are represented by the cross and circle marks.

Table 1 Solutions used in the experiments by Hodgkin and Horowicz.

K^+	Na^+	Ca^{2+}	HPO_4^{2-}	$H_2PO_4^{2-}$	SO_4^{2-}
←		(mM)			→
0 ~ 80	83 ~ 3	8	1.08	0.43	48

* These solutions have the same tonicity and ionic strength as Ringger's fluid.

Even though the accuracy of the equations seems to be confirmed in Figure 4, it can also be thought as if it is a coincidence, since $[K]_i$, assumed as constant and equal to 140mM, also changes when there is change in the concentration of extracellular K^+. Further more, the point O– at left in Figure 4 cannot be explained by both Equations (17) and (18).

Using Nernst relation, from Equations (1) and (12), the following equation can be obtained:

$$V_m = \frac{RT}{z_X F} \ln \frac{[X]_o}{[X]_i} - \frac{I_{P,X}}{g_X} \quad (19)$$

If value of $I_{P,K}/g_K$ is small enough, Equation (19) can be written as Equation (18), and it becomes correct in the beginning of change of $[K]_o$. The gap between the line and the experimental results in Figure 4, thought as being caused by $[Na]_o$, can be considered as caused by variation of $[K]_i$. Since the lower the $[K]_o$, the faster the intracellular K^+ flows out through the membrane.

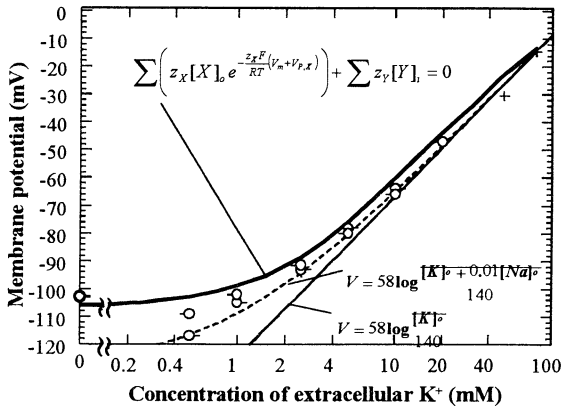


Figure 4. Relation between membrane potential and $\log[K]_o$, when using chloride-free sulphate solution containing 8 mM-$CaSO_4$ as described in Table 1. Estimated results (thick line) are plotted from Equation (14).

Table 2 Concentrations of intracellular and extrcellular ions of frogs muscle fibers. (Chuma, 1971; Boyle et al., 1941)

ion	Intracellular Concentration (mM)	Extracellular Concentration (mM)
K^+	124.0	2.3
Na^+	10.4	109.0
Cl^-	1.5	77.5
Ca^{2+}	4.9	2.1
Mg^{2+}	14.0	1.3
HCO_3^-	12.4	26.6
A^-	74.0	13.0
Resting membrane potential V_m		71mV

We can estimate the resting membrane potential using Equation (14) under the same conditions given in Table 1. Then, from Equations (12) and (1), active transport potential can be obtained as

$$V_{AT,X} = \frac{I_{P,X}}{g_X} = -\left(V_m + \frac{RT}{z_X F}\ln\frac{[X]_i}{[X]_o}\right) \tag{20}$$

Table 2 shows the intra and extracellular concentrations of frog muscle fibers. Using Table 2, the following active transport potentials of frogs muscle fibers in steady state can be obtained from Equation (20):

$$\left.\begin{array}{ll} V_{AT,Na} = 0.149\,\mathrm{mV}, & V_{AT,K} = 0.009\,\mathrm{mV}, \\ V_{AT,Cl} = 0.0\,\mathrm{mV}, & V_{AT,Ca} = 0.079\,\mathrm{mV}, \\ V_{AT,Mg} = 0.059\,\mathrm{mV}, & V_{AT,HCO3} = 0.071\,\mathrm{mV} \end{array}\right\} \tag{21}$$

The above results show that Ca^{+2}, Mg^{+2}, and HCO_3^{-1} also have active transport, and K^+ has a very weak active transport potential due to its big conductance. In this paper we do not discuss the physiological principles of active transport.

Assuming that active transport potentials are the same in any steady state[†1], using Equation (21) and Table 1, from Equation (14) a relation between membrane potential and $[K]_o$, shown in Figure 4 (thick line), can be obtained.

Figure 4 shows that the estimated results (thick line) using Equation (14) are similar to the experimental ones, including the case of $[K]_o$=0, which can not be calculated by GHK equation. The errors between the estimated results (thick line) and the experimental ones can be thought as caused by: (a) change of active transport potentials due to change of $[K]_o$, (b) cells were not in real steady state, i.e., in the experiments, there was not enough time to let the cells get to the steady state, and (c) concentration of intracellular ATP became lower than normal since the cells were taken out from living body for a long time.

[†1] The mechanism of active transport is very complex. The transport ability is affected by many conditions, such as intra and extracellular ion concentrations, membrane potential, and concentration of ATP. We assume that if there exist enough ATP in a cell, the active transport potentials turn back to the standard levels when the cell reaches the steady state after a change of extracellular ion concentrations.

6. Conclusion

This paper proposed a new equivalent circuit of cell by introducing the concepts of *ion-condensers* and current sources. Using this equivalent circuit, we obtained an equation for the resting membrane potential, and proved the influence of impermeant anions concentrations and ion-pumps on it. The results are the following:

1. The resting membrane potential of a cell is determined by
   (a) concentrations of extracellular ions grouped by valences;
   (b) intracellular impermeant anions concentrations;
   (c) the ratio of active transport current and equivalent conductance of permeability through the membrane of each ion, i.e., active transport potential; and
   (d) temperature around the cell.
2. GHK equation is not suitable for liveing bio-membranes, and resting membrane potential can be estimated using the equation proposed in the present paper.

Acknowledgements

This study was supported by Grant-in-Aid for Encouragement of Young Scientists of the Ministry of Education.

7. References

[1] Alberts B. et al. (1994). Molecular biology of the cell, 3rd Edition, Garland Publishing, Inc., New York

[2] Boyle P. J., E. J. and Conway (1941). Potassium accumulation in muscle and associated changes, J. Physiol. 100, 1-63.

[3] Hall, ZW. (1992). Molecular Neurobiology, Sinauer Associates. Inc., New York

[4] Hodgkin, AL. (1962). Ionic movements and electrical activity in giant nerve fibres. The Croonian Lecture, Proc. Roy. Soc. B, vol. 148, 1-37.

[5] Hodgkin, AL., and AF., Huxley, (1952). A Quantitative Description of Membrane Current and Its Application to Conduction and Excitation in Nerve. J. Physiol. 117, 500-544.

[6] Hodgkin, AL., and AF. Huxley, (1952). Currents Carried by Sodium and Potassium Ions Through The Membrane of The Giant Axon of Loligo. J. Physiol. 116, 449-472.

[7] Hodgkin, AL and P. Horowicz, (1959). The Influence of potassium and chloride ions on the membrane potential of single muscle fibres. J. Physiol. 148, 127-160.

[8] Chuma, I. (1971). Physiology, Japan Medical Journal Inc., Tokyo.

[9] Zhang, X., and H.Wakamatsu, (1999): Electrical equivalent circuit and resting membrane potential of neuron. Proc. 4th Int. Symp. on Artificial Life and Robotics, 703-706.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A Simple Robust Tracking Controller for Robot Manipulators Using Only Joint Position Measurements

Makoto Wada*, Masahiro Oya*, Ryozo Katoh*, and Hideki Honda**

* Department of Mechanical and Control Engineering Faculty of Engineering
Kyushu Institute of Technology, Tobata, Kitakyushu, 804-8550, Japan
E-mail: Makoto@postman.cntl.kyutech.ac.jp
** Research Laboratory YASKAWA ELECTORIC CORPORATION
YahataNishi, Kitakyushu, 806-0004, Japan
E-mail: hideki@yaskawa.co.jp

Abstract

In this paper we develop a new robust trajectory tracking control scheme without using joint velocity. The proposed controller doesn't employ adaptation. Therefore, the construction of the controller becomes very simple. Moreover, it is shown that we can make the Euclid norm of tracking errors arrive to any small closed region with any convergent rate.

1. Introduction

Many researchers have studied tracking control of robotic manipulators and several schemes requiring full state measurements have been developed. However, in the velocity measurement of joints, there exist the problem that the required accuracy can not achieved in practical applications due to the existence of noises. Therefore, tracking controllers without using velocity measurements have been proposed [1]-[3]. The controllers proposed in [1]-[3] require full knowledge of robot dynamics. If there are uncertainties in the dynamics, the controllers may given poor performance and may cause instability. To resolve the problem, the robust control schemes using regressor based adaptive controllers were developed [4]-[10]. The adaptive controllers are effective for uncertainties in robot manipulators, and guarantee asymptotical stability of the tracking errors. However, the controllers become very complex and are not suitable for real-time implementation. An additional observation is that the main attention in [4]-[10] is focused on stability analysis.

In the paper, we develop a new robust tracking control scheme without using adaptations and velocity measurements based on a state observer. The developed robust controller becomes very simple compared with that of the regressor based adaptive controllers. It is shown that the Euclid norm of tracking errors can be arbitrarily reduced by using only one design parameter. Especially for the ideal trajectories q_d such that $\dot{q}_d$ converge to zero exponentially, it is shown that Euclid norm of tracking errors converges to zero.

2. Robot Dynamics and Tracking Error Equation

The equation of motion for n-link rigid robot is given by

$$M(q)\ddot{q}(t) + C(q,\dot{q})\dot{q} + G(q) = u(t), \quad (1)$$

where $q(t) \in R^n$ is the joint angle vector. $M(q) \in R^{n\times n}$ is the positive definite inertia matrix of manipulator, $C(q,\dot{q}) \in R^{n\times n}$ is the centripetal and Coriolis terms, and $G(q) \in R^n$ is the gravity terms. $u(t) \in R^n$ is the motor torques.

It is well known that manipulator is characterized by following properties; **P1.** The inertia and the centripetal and Coriolis matrix satisfy the following relationship $x^T(\dot{M}(q) - 2C(q,\dot{q}))x = 0 \quad \forall x \in R^n$, **P2.** The centripetal and Coriolis matrix satisfies the following relation $C(q,x)y = C(q,y)x \ \forall x,y \in \mathrm{R}^n$, **P3.** The matrix $M(q)$ is symmetric positive definite and there exist bounded constant values $\underline{\rho}_1, \bar{\rho}_1$ such that $\underline{\rho}_1 x^T x \le x^T M(q) x \le \bar{\rho}_1 x^T x$ for any vector x, **P4.** There exist positive constants $\bar{\rho}_2$, $\bar{\rho}_3$ such that $\|C(q,\dot{q})\| \le \bar{\rho}_2 \|\dot{q}(t)\|$, $\|G(q)\| \le \bar{\rho}_3$ respectively.

In this paper, it is assumed that only the joint angle $q(t)$ measurement is available. Under the assumption, let's consider a controller given by the form of

$$u(t) = u_1(t) + \hat{M}(q)\ddot{q}_d(t) + \hat{C}(q,\dot{q}_d)\dot{q}_d(t) + \hat{G}(q), \quad (2)$$

where the terms $\hat{M}(q)$, $\hat{C}(q,\dot{q}_d)$ and $\hat{G}(q)$ are estimates of $M(q)$, $C(q,\dot{q})$ and $G(q)$ respectively. q_d is the desired trajectory. Substituting $u(t)$ in (2) into (1), we obtain

$$\begin{aligned} &M(q)\ddot{\tilde{q}}(t) + C(q,\dot{\tilde{q}})\dot{\tilde{q}} + 2C(q,\dot{q}_d)\dot{\tilde{q}}(t) \\ &= u_1(t) - \tilde{M}(q)\ddot{q}_d(t) - \tilde{C}(q,\dot{q}_d)\dot{q}_d(t) - \tilde{G}(q) \end{aligned} \quad (3)$$

where

$$\left.\begin{aligned} &\tilde{q}(t) = q(t) - q_d(t), \ \tilde{C}(q,\dot{q}_d) = C(q,\dot{q}_d) - \hat{C}(q,\dot{q}_d) \\ &\tilde{M}(q) = M(q) - \hat{M}(q), \ \tilde{G}(q) = G(q) - \hat{G}(q) \end{aligned}\right\} \quad (4)$$

To make $q(t)$ track a desired trajectory $q_d(t)$ such that $\dot{q}_d$ converges to zero exponentially, the pre-compensator $u_1(t) = \mu_1(t) + d\int_0^t \mu_1(\tau)d\tau$ is introduced. The signal $\mu_1(t)$ is the input signal to the pre-compensator and the design parameter d is positive constant. Then, the equation (3) is rewritten as (See Appendix A).

$$\begin{aligned} &M(q)\dddot{\tilde{q}}(t) + C(q,\dot{\tilde{q}})\ddot{\tilde{q}} + 2C(q,\dot{q}_d)\ddot{\tilde{q}}(t) - dM(q)\ddot{\tilde{q}}(t) \\ &+ (M(q) - \hat{M}(q))\dddot{q}_d(t) + (C(q,\dot{q}_d) - \hat{C}(q,\dot{q}_d))\ddot{q}_d(t), \\ &+ d\beta^2 z_1 + d\beta z_2 + \beta z_3 - \beta z_4 - \phi = \mu_1 \end{aligned} \quad (5)$$

$$\dot{z}_1 = -dz_1 + \beta^{-2}(\dot{M}(q) - C(q,\dot{\tilde{q}})\dot{\tilde{q}}(t), \quad (6)$$

$$\dot{z}_2 = -dz_2 + \beta^{-1}(dM(q) - 2C(q,\dot{q}_d))\dot{\tilde{q}}(t), \quad (7)$$

$$\dot{z}_3 = -dz_3 + \beta^{-1}\frac{\partial \tilde{G}(q)}{\partial q}\dot{\tilde{q}}(t), \quad (8)$$

$$z_4 = -dz_4 + \beta^{-1}\tilde{M}(q)\dot{q}_d + \beta^{-1}\tilde{C}(q,\dot{q}_d)\dot{q}_d + \beta^{-1}\frac{\partial \tilde{G}(q)}{\partial q}\dot{q}_d(t), \quad (9)$$

$$\phi(t) = \exp(-dt)\left(dM(q(0))\dot{\tilde{q}}(0) + \tilde{G}(q(0))\right), \quad (10)$$

where $\beta\,(>2/d)$ is positive design parameter. It should be noted that the gravitational torques $\tilde{G}(q)$ disappears in the new presentation above. Instead of $\tilde{G}(q)$ the term $\partial\tilde{G}(q)/\partial q\dot{\tilde{q}}(t)$ and $\partial\tilde{G}(q)/\partial q\dot{q}_d(t)$ appear. It can be seen from the fact that if we can develop tracking controller for the new representation above, the tracking error converges to zero in the case of constant desired trajectories.

3. Controller Development

To develop a controller, the lemma 1 below is shown and the following assumption is made for the robot model (5)-(10); **P5.** There exist positive constant $\bar{\rho}_{it}\ (i=1,\cdots,4)$ $\bar{\rho}_4$, such that, $\|\tilde{M}(q)\| \le \bar{\rho}_{1t}$, $\|\tilde{C}(q,\dot{q})\| \le \bar{\rho}_{2t}\|\dot{q}(t)\|$, $\|\partial\tilde{G}(q)/\partial q\| \le \bar{\rho}_{3t}$, $\|\tilde{G}(q)\| \le \bar{\rho}_{4t}$, $\|\dot{M}(q)\| \le \bar{\rho}_4\|\dot{q}(t)\|$ respectively, **P6.** For any given desired trajectories $q_d(t)$ there exist bounded positive constant values $\bar{d}_0, \bar{d}_1, \bar{d}_2$ as follows $\|q_d(t)\| \le \bar{d}_0$, $\|\dot{q}_d(t)\| \le \bar{d}_1$, , $\|\ddot{q}_d(t)\| \le \bar{d}_2$, and the following lemma 1 is shown.

Lemma 1: Let $V(t) \in R$ be positive and satisfies

$$\dot{V}(t) \le -dV(t) + \beta^{-1}V(t)^2 + \beta^{-1}\varepsilon, |\varepsilon| \le 1, \beta(\ge 2/d) \quad (11)$$

(1) When $\varepsilon \ne 0$ and the initial value of $V(0)$ is given by

$$V(0) \le D_-, D_- = 2\beta\left(d - \sqrt{d^2 - 4\beta^{-2}}\right) \quad (12)$$

$V(t)$ satisfies $V(t) \le D_-$.

(2) When $\varepsilon \ne 0$ and the initial value of $V(t)$ is given by

$$D_- < V(0) \le D_+, D_+ = 2\beta\left(d + \sqrt{d^2 - 4\beta^{-2}}\right) \quad (13)$$

$V(t)$ satisfies $V(t) \le V(0)$.

(3) When $\varepsilon = 0$ and $V(0) \le D_+$, then $\lim_{t\to\infty} V(t) = 0$.

Proof: The proof of lemma 1 is omitted for rack of space.

If joint velocity is available, we can easily prove robust stability of the system using the simple controller

$$\mu_1(t) = -\beta\left(f_s(\beta) + \alpha\bar{\rho}_1\right)s(t), s(t) = \beta^{-1}\dot{\tilde{q}}(t) + \tilde{q}(t)$$
$$f_s(\beta) = \beta^3\bar{\rho}_{s1} + \beta^2\bar{\rho}_{s2} + \beta\bar{\rho}_{s3} + \bar{\rho}_{s4} + d\bar{\rho}_1 \quad (14)$$

where α is positive design parameter introduced to improve the tracking performance and $\bar{\rho}_{si}$ are positive constant specified later. However, in fact, the tracking error signal $s(t)$ is not available. Therefore, let's consider the following observer to estimate the error signal $s(t)$.

$$\left.\begin{aligned}\hat{s}(t) &= \beta^{-1}(l_1(t) + \gamma_e\tilde{q}(t)) + \tilde{q}(t)\\ \dot{l}_1(t) &= -\gamma_e(l_1(t) + \gamma_e\tilde{q}(t)), \quad l_1(0) = \beta^{-1}\gamma_e\tilde{q}(0) + \tilde{q}(0)\end{aligned}\right\} \quad (15)$$

The parameter γ_e is the gain given by the form of

$$\left.\begin{aligned}\gamma_e &= \frac{1}{2\underline{\rho}_1}\left(f_s(\beta) + f_e(\beta) + 2\alpha\right)\\ f_e(\beta) &= \beta^3\bar{\rho}_{e1} + \beta^2\bar{\rho}_{e2} + \beta\bar{\rho}_{e3} + \bar{\rho}_{e4} + d\underline{\rho}_1\end{aligned}\right\} \quad (16)$$

where $\bar{\rho}_{ei}$ are positive constant specified later.

Using the controller with estimated signal

$$\mu_1(t) = -\beta\left(f_s(\beta) + \alpha\right)\hat{s}(t), \quad (17)$$

the following theorem holds. Before showing this, we define the following parameters and the positive functions.

$$\left.\begin{aligned}\delta_q &= \max\left(\begin{matrix}\sqrt{\frac{\sqrt{29}}{20}}(\bar{\rho}_4 + \bar{\rho}_2), \frac{8\sqrt{29}}{5}\beta^{-2}\frac{\bar{\rho}_4^2\bar{d}_1^2}{\bar{\rho}_4 + \bar{\rho}_2},\\ 2\beta^{-1}\bar{\rho}_1, \frac{8\bar{\rho}_2\bar{d}_1^2}{5}, \frac{64\beta^{-1}\bar{\rho}_{3t}}{d}\end{matrix}\right)\\ \bar{\rho}_{h1} &= 20(\bar{\rho}_4 + \bar{\rho}_2),\ \bar{\rho}_{h2} = \max\left(20d\bar{\rho}_1, \bar{\rho}_2\bar{d}_1, 20\bar{\rho}_2\right)\\ \bar{\rho}_{h3} &= \bar{\rho}_{3t},\ \bar{\rho}_{h4} = \frac{80\beta^{-1}\bar{d}_1^2}{d} \times \max\left(\bar{\rho}_{1t}^2, \bar{\rho}_{2t}^2, \bar{\rho}_{3t}^2\right)\end{aligned}\right\}, \quad (18)$$

$$\left.\begin{aligned}V &= V_{s\tilde{s}} + V_{\tilde{q}} + V_h + V_\phi\\ V_{s\tilde{s}} &= V_s + V_{\tilde{s}}, V_s = s^T M s, V_{\tilde{s}} = \tilde{s}^T M\tilde{s}, V_\phi = \phi^T\phi\\ V_q &= \delta_q\tilde{q}^T\tilde{q}, V_h = h_i^T h_i, h_i = \frac{1}{\sqrt{\bar{\rho}_{hi}}}z_i\end{aligned}\right\}. \quad (19)$$

Theorem 1: Let the design parameter d be set to be a fixed constant value. Then, if the design parameter β is set so that the relation $V(0) \le D_+$ is satisfied and $\bar{\rho}_{si}, \bar{\rho}_{ei}$ are set so as to satisfy the following relation.

$$\left.\begin{aligned}\bar{\rho}_{s1} &\ge \frac{3\bar{\rho}_2^2}{\underline{\rho}_1\delta_q} + \frac{4\bar{\rho}_2^2}{\delta_q^2}, \bar{\rho}_{s2} \ge 2d\rho_{h1}\\ \bar{\rho}_{s3} &\ge 18\bar{\rho}_1 + \frac{4\bar{\rho}_2^2\bar{d}_1^2}{\underline{\rho}_1} + \frac{16\bar{\rho}_2^2\bar{d}_1^2}{\delta_q} + \frac{\sqrt{2}\bar{\rho}_2^2\bar{d}_1^2}{10}\\ &\quad + \frac{\sqrt{29}(\bar{\rho}_2 + \bar{\rho}_4)}{5\delta_q} + \frac{\sqrt{29}(\bar{\rho}_2 + \bar{\rho}_4)}{20\underline{\rho}_1} + 2\delta_q\\ \bar{\rho}_{s4} &\ge 3d\bar{\rho}_1 + 16\beta^{-1}d^2\bar{\rho}_1 + 16\beta^{-1}\bar{\rho}_{1t}^2\bar{\rho}_2^2\\ &\quad + 16\beta^{-1}\bar{\rho}_{2t}^2\bar{d}_1^4 + 5d\bar{\rho}_{h2} + \frac{\sqrt{29}\beta^{-1}\bar{\rho}_4\bar{d}_1^2}{10} + \frac{\bar{\rho}_1}{4}\\ &\quad + \frac{\sqrt{29}\beta^{-1}\bar{\rho}_4\bar{d}_1^2}{10} + \frac{4\bar{\rho}_{h3} + 5\bar{\rho}_{h4} + 2\beta^{-1} + 4\bar{\rho}_{3t}}{d}\\ \bar{\rho}_{e1} &\ge \frac{4\bar{\rho}_2^2}{\underline{\rho}_1^2} + \frac{3\bar{\rho}_2^2}{\underline{\rho}_1\delta_q} + \frac{4\bar{\rho}_2^2}{\delta_q^2} + \frac{\bar{\rho}_4^2}{4\underline{\rho}_1^2} + \frac{\bar{\rho}_4^2}{4\underline{\rho}_1\delta_q}\\ \bar{\rho}_{e2} &\ge 2d\bar{\rho}_{h1},\ \bar{\rho}_{e3} \ge \frac{2\left(2\bar{\rho}_2^2 + \bar{\rho}_4^2\right)\bar{d}_1^2}{\underline{\rho}_1} + \frac{4\bar{\rho}_2^2\bar{d}_1^2}{\delta_q}\\ \bar{\rho}_{e4} &\ge d\bar{\rho}_1 + 16\beta^{-1}\left(d^2\bar{\rho}_1^2 + \bar{\rho}_{1t}^2\bar{d}_2^2 + \bar{\rho}_{2t}^2\bar{d}_1^4\right)\\ &\quad + 16d\bar{\rho}_{h2} + \frac{4\bar{\rho}_{h3} + 5\bar{\rho}_{h4} + 2\beta^{-1}}{d}\end{aligned}\right\} \quad (20)$$

The closed loop system become stable, and the tracking error $\tilde{q}(t)$ satisfies

$$\|\tilde{q}(t)\|^2 \le \bar{c}_1\exp(-\beta t) + \frac{\bar{c}_2}{\alpha}, \quad (21)$$

where $\bar{c}_i$ are positive constant values independent of design parameter α.

Proof: The positive function V satisfies (See Appendix B)

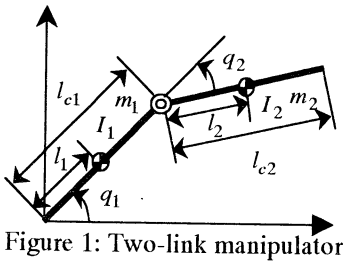


Figure 1: Two-link manipulator

$$\left.\begin{aligned}\dot{V} &\le -dV + \beta^{-1}V^2 + \beta^{-1}\varepsilon - \alpha V_{s\tilde{s}} \\ &\le -dV + \beta^{-1}V^2 + \beta^{-1}\varepsilon \\ \varepsilon &= \frac{1}{16}\left\{15\frac{\|\dot{q}_d\|^2}{\bar{d}_1^{\,2}} + \frac{\|\ddot{q}_d\|^2}{\bar{d}_2^{\,2}}\right\} \le 1\end{aligned}\right\} \quad (22)$$

According to lemma 1, it can be seen from (22) that closed loop system becomes stable.

Next, we show that the relation (21) holds. The positive function $V_{s\tilde{s}}$ satisfies (See Appendix B)

$$\dot{V}_{s\tilde{s}} \le -\alpha V_{s\tilde{s}} + \beta^{-1}V^2 + \varepsilon_1 \quad (23)$$

where ε_1 is positive constant value independent of design parameter α. From (23) it can be seen that the relation (21) holds. ■

In the controller (17) and observer (15), only scalar feedback gains and design parameters are used. The construction of proposed controller is very simple compared to that of adaptive controller. From the relation (21) it can be concluded that the tracking performance is improved by only one design parameter α. For the ideal trajectory q_d such that $\dot{q}_d$ converges to zero exponentially, in addition to theorem 1, the following theorem also holds.

Theorem 2: We consider ideal trajectory such that $\|\dot{q}_d\| \le \bar{d}_1 e^{-at}$ and $\|\ddot{q}_d\| \le \bar{d}_2 e^{-at}$. Let the design parameter d be set to be a fixed positive constant value ($d \le a$) and the design parameter β be set so that the relation $V(0) + \frac{1}{a\beta} \le D_+$ is satisfied and the relations (20) be satisfied. Then, the tracking error $\tilde{q}$ converses to zero.

Proof: Analyzing the derivative of the positive function $V_1 = V + \frac{1}{a\beta}\phi_d{}^T\phi_d$, where $\dot{\phi}_d = -a\phi_d$, $\phi_d(0) = 1$, it can be seen that the following relation is satisfied.

$$\dot{V}_1 \le -dV_1 + \beta^{-1}V_1^2 . \quad (24)$$

From lemma 1 and (24) it can be seen that tracking error $\tilde{q}$ converse to zero. ■

4. Simulation example

As an illustration, we apply the proposed controller (17) to a two-link robot manipulator shown in Fig 1. The value of the manipulator are set as; l_1 = 0.3m, m_1 = 11.1kg, l_{c1} = 0.159m, I_1 = 0.869kgm^2, l_2 = 0.3m, m_2 = 4.05kg, l_{c2} = 0.0537m, I_2 = 0.052kgm^2, and the estimated parameter of manipulator are set as; $\hat{l}_1$ = 0.25m, $\hat{m}_1$ = 10kg, $\hat{l}_{c1}$ = 0.1m, $\hat{I}_1$ = 0.75kgm^2, $\hat{l}_2$ = 0.25m, $\hat{m}_2$ = 3.5kg, $\hat{l}_{c2}$ = 0.04m, $\hat{I}_2$ = 0.04kgm^2. The design parameter is set as $d = 1, \beta = 5$. Fig.2 and Fig.3 show the desired trajectories for link1 and link2. Fig.4 shows simulation results for ideal trajectory I. Fig.4 (a) shows the joint angle tracking errors of link1 for various design parameters α and (b) shows control input signals Fig.5 shows simulation results for ideal trajectory II. Fig.5. (a) shows the joint angle tracking errors of link1 for various design parameter α and (b) shows control input signals. As shown in Fig.4, for the ideal trajectories q_d such that $\dot{q}_d$ converges to zero exponentially, the tracking error converges to zero. As shown in Fig4 and Fig.5 the tracking performance can be improved by the design parameter α.

5. Conclusion

In this paper, we have developed a new robust controller using only joint position measurement based on an observer to obtain estimates of the joint velocity signal. The controller ensures that the Euclid norm of tracking errors can arrive to any small closed region with any convergent rate by setting only one design parameter. Especially for the ideal trajectory q_d such that $\dot{q}_d$ converges to zero exponentially, it is shown that the Euclid norm of tracking errors converges to zero.

Appendix A Now, let's consider the signal given by

$$\left.\begin{aligned}x_{a1} &= \phi(t) + M(q)\dot{\tilde{q}}(t) + d^{-1}\tilde{G}(q) \\ &\quad + \int_0^t e^{-d(t-\tau)}\left(M(q)\ddot{\tilde{q}}(t) + C(q,\tilde{q})\dot{\tilde{q}}(t) + 2C(q,\dot{q}_d)\dot{\tilde{q}}(t)\right. \\ &\quad \left. + \tilde{M}(q)\ddot{q}_d(t) + \tilde{C}(q,\dot{q}_d)\dot{q}_d(t) + \tilde{G}(q)\right)d\tau \\ x_{a2} &= \int_0^t e^{-d(t-\tau)}u_1(t)d\tau\end{aligned}\right\} \quad (a1)$$

Then, from (3) it can be seen that the relation $\dot{x}_{a1} = \dot{x}_{a2}$ satisfied. It is seen from $\dot{x}_{a1} = \dot{x}_{a2}$ that using the relation $u_1(t) = \mu_1(t) + d\int_0^t \mu_1(\tau)d\tau$, the model (5)-(10) can be derived.

Appendix B The positive function V_h satisfies

$$\begin{aligned}\dot{V}_h &\le -2dh_1{}^Th_1 - 2dh_2{}^Th_2 - 2dh_3{}^Th_3 - 2dh_4{}^Th_4 \\ &\quad + (\beta\rho_{ha} + \rho_{hb})\|s\|^2 + \beta^{-1}\frac{7}{16}\frac{\|\dot{q}_d\|^2}{\bar{d}_1^{\,2}} \\ &\quad + \left(\frac{5}{16}\beta\delta_q\|\tilde{q}\|^2 + \frac{3}{5}d\|h_2\|^2 + \frac{3}{4}d\|h_3\|^2 + \frac{3}{5}d\|h_4\|^2\right) \\ &\quad + \beta^{-1}\left(\frac{1}{4}\underline{\rho}_1{}^2\|s\|^4 + \frac{1}{2}\delta_q{}^2\|\tilde{q}\|^4 + \|h_2\|^4\right).\end{aligned} \quad (b1)$$

The positive function $V_{s\tilde{s}}$ satisfies

$$\begin{aligned}\dot{V}_{s\tilde{s}} &\le -d\bar{\rho}_1 s^Ts - d\bar{\rho}_1\tilde{s}^T\tilde{s} - \alpha\bar{\rho}_1 s^Ts - \alpha\bar{\rho}_1\tilde{s}^T\tilde{s} \\ &\quad -\left(\beta\rho_{ha} + \rho_{hb} + 2\delta_q\right)\|s\|^2 + \frac{1}{2}\beta\delta_q\|\tilde{q}\|^2 \\ &\quad + \beta^{-1}\left(\frac{3}{4}\underline{\rho}_1{}^2\|s\|^4 + \frac{1}{2}\delta_q{}^2\|\tilde{q}\|^4 + \underline{\rho}_1\delta_q\|s\|^2\|\tilde{q}\|^2 + \|h_1\|^4\right. \\ &\quad \left. + \underline{\rho}_1{}^2\|s\|^2\|\tilde{s}\|^2 + \underline{\rho}_1\delta_q\|\tilde{s}\|^2\|\tilde{q}\|^2 + \underline{\rho}_1{}^2\|\tilde{s}\|^4\right) \\ &\quad + \beta^{-1}\frac{1}{16}\left\{8\frac{\|\dot{q}_d\|^2}{\bar{d}_1^{\,2}} + \frac{\|\ddot{q}_d\|^2}{\bar{d}_2^{\,2}}\right\} \\ &\le -\alpha V_{s\tilde{s}} + \beta^{-1}V^2 + \varepsilon_1 ,\end{aligned} \quad (b2)$$

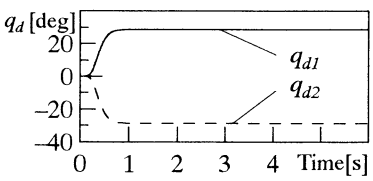


Figure 2: Ideal trajectory I

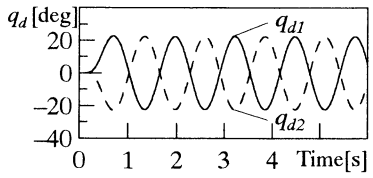
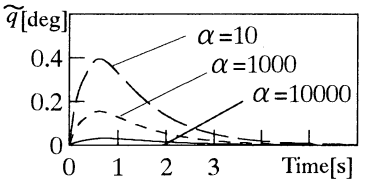
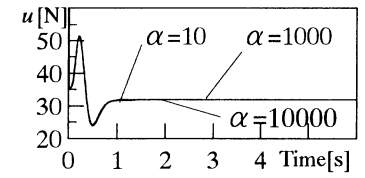


Figure 3: Ideal trajectoryII

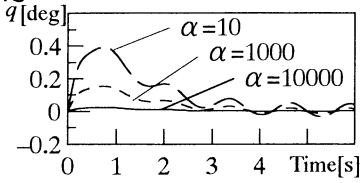


(a) Position tracking error of joint 1

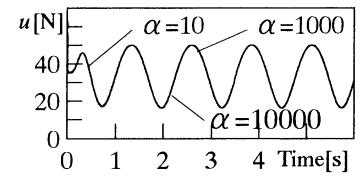


(b) Input to joint 1

Figure 4: Response for the ideal trajectory I



(a) Position tracking error of joint 1



(b) Input to joint 1

Figure 5: Response for the ideal trajectoryII

where

$$\left.\begin{aligned}\bar{\rho}_{ha} &= \frac{\sqrt{29}(\bar{\rho}_2+\bar{\rho}_4)}{20}\left\{\frac{1}{\underline{\rho}_1}+\frac{4}{\delta_q}\right\}+\frac{\sqrt{2}\bar{\rho}_2\bar{d}_1^{\,2}}{10}\\ \bar{\rho}_{hb} &= \frac{\beta^{-1}\sqrt{29}\bar{\rho}_4\bar{d}_1^{\,2}}{10}+\frac{4\bar{\rho}_3}{d}\end{aligned}\right\} \quad \text{(b3)}$$

In the derivation of (b1), (b2), the following relations are used. In the relations below, the same relation is satisfied when s is represented with $\tilde{s}$, the same relation are also used to derive (b1), (b2).

$$2\beta a_1\|s\|\|\tilde{q}\| \le 16\beta a_1\|s\|^2+\frac{1}{16}\beta a_1\|\tilde{q}\|^2$$

$$2\beta a_1\|s\|^2\|\tilde{q}\| \le 3\beta^3\frac{a_1^{\,2}}{\delta_q\underline{\rho}_1}\|s\|^2+\beta^{-1}\frac{1}{3}\delta_q\underline{\rho}_1\|s\|^2\|\tilde{q}\|^2$$

$$2\beta a_1\|s\|\|\tilde{q}\|^2 \le 4\beta^3\frac{a_1^{\,2}}{\delta_q^{\,2}}\|s\|^2+\beta^{-1}\frac{1}{4}\delta_q^{\,2}\|\tilde{q}\|^4$$

$$2\beta a_1\|s\|\|\tilde{s}\| \le \beta a_1\|s\|^2+\beta a_1\|\tilde{s}\|^2$$

$$2\beta^{-1}a_1\|s\|\|\ddot{q}_d\| \le 16\beta^{-1}a_1^{\,2}\bar{d}_2^{\,2}\|s\|^2+\beta^{-1}\frac{\|\ddot{q}_d\|^2}{16\bar{d}_2^{\,2}}$$

$$2\beta^{-1}a_1\|s\|\|\dot{q}_d\|^2 \le 16\beta^{-1}a_1^{\,2}\bar{d}_1^{\,4}\|s\|^2+\beta^{-1}\frac{\|\dot{q}_d\|^2}{16\bar{d}_1^{\,2}}$$

$$2a_1\|s\|^2\|\dot{q}_d\| \le \frac{4\beta a_1^{\,2}\bar{d}_1^{\,2}}{\underline{\rho}_1}\|s\|^2+\beta^{-1}\frac{1}{4}\underline{\rho}_1^{\,2}\|s\|^4+\beta^{-1}\frac{\|\dot{q}_d\|^2}{16\bar{d}_1^{\,2}}$$

$$2a_1\|s\|\|\tilde{q}\|\|\dot{q}_d\| \le \frac{4\beta a_1^{\,2}\bar{d}_1^{\,2}}{\delta_q}\|s\|^2+\beta^{-1}\frac{1}{4}\delta_q^{\,2}\|\tilde{q}\|^4+\beta^{-1}\frac{\|\dot{q}_d\|^2}{16\bar{d}_1^{\,2}}$$

$$2d\beta\sqrt{\bar{\rho}_{h1}}\|s\|\|h_1\| \le 2\beta^2 d\bar{\rho}_{h1}\|s\|^2+\frac{1}{2}d\|h_1\|^2$$

$$2a_1\sqrt{\bar{\rho}_{hi}}\|s\|\|h_i\| \le \frac{c_i a_1^{\,2}\bar{\rho}_{hi}}{d}\|s\|^2+\frac{1}{c_i}d\|h_i\|^2$$

$$\frac{2}{\sqrt{\bar{\rho}_{h1}}}a_1\|h_1\|\|s\|^2 \le \frac{\sqrt{29}}{\underline{\rho}_1\bar{\rho}_{h1}}\beta a_1^{\,2}\|s\|^2+\frac{1}{29}\beta^{-1}\|h_1\|^4+\frac{1}{4}\beta^{-1}\underline{\rho}_1^{\,2}\|s\|^4$$

$$\frac{2}{\sqrt{\bar{\rho}_{h1}}}a_1\|h_1\|\|s\|\|\tilde{q}\| \le \frac{\sqrt{29}}{\delta_q\bar{\rho}_{h1}}\beta a_1^{\,2}\|s\|^2+\frac{1}{29}\beta^{-1}\|h_1\|^4+\frac{1}{4}\beta^{-1}\delta_q^{\,2}\|\tilde{q}\|^4$$

$$\frac{2}{\sqrt{\bar{\rho}_{h1}}}a_1\|h_1\|\|\tilde{q}\|^2 \le \frac{\sqrt{29}}{\delta_q\bar{\rho}_{h1}}\beta a_1^{\,2}\|\tilde{q}\|^2+\frac{1}{29}\beta^{-1}\|h_1\|^4+\frac{1}{4}\beta^{-1}\delta_q^{\,2}\|\tilde{q}\|^4$$

$$\frac{2}{\sqrt{\bar{\rho}_{hj}}}\beta^{-1}a_1\|h_1\|\|s\|\|\dot{q}_d\| \le \frac{2\sqrt{29}}{\bar{\rho}_{hj}}\bar{d}_1^{\,2}a_1\beta^{-1}\|s\|^2+\frac{1}{29}\beta^{-1}\|h_1\|^4+\frac{1}{16}\beta^{-1}\frac{\|\dot{q}_d\|^2}{\bar{d}_1^{\,2}}$$

$$\frac{2}{\sqrt{\bar{\rho}_{hj}}}\beta^{-1}a_1\|h_1\|\|\tilde{q}\|\|\dot{q}_d\| \le \frac{2\sqrt{29}}{\bar{\rho}_{hj}}\bar{d}_1^{\,2}a_1^{\,2}\beta^{-1}\|\tilde{q}\|^2+\frac{1}{29}\beta^{-1}\|h_1\|^4+\frac{1}{16}\beta^{-1}\frac{\|\dot{q}_d\|^2}{\bar{d}_1^{\,2}}$$

$$\frac{2}{\sqrt{\bar{\rho}_{hk}}}a_1\|h_2\|\|s\| \le \frac{c_k}{\bar{\rho}_{hk}d}a_1^{\,2}\|s\|^2+\frac{1}{c_k}d\|h_2\|^2$$

$$\frac{2}{\sqrt{\bar{\rho}_{hk}}}a_1\|h_2\|\|\tilde{q}\| \le \frac{c_k}{\bar{\rho}_{hk}d}a_1^{\,2}\|\tilde{q}\|^2+\frac{1}{c_k}d\|h_2\|^2$$

$$\frac{2}{\sqrt{\bar{\rho}_{h4}}}\beta^{-1}a_1\|h_4\|\|\dot{q}_d\| \le \frac{16a_1^{\,2}}{\bar{\rho}_{h4}}\beta^{-1}\|h_4\|^2+\beta^{-1}\frac{1}{16}\frac{\|\dot{q}_d\|^2}{\bar{d}_1^{\,2}}$$

where a_1 is constant value. $c_2 = c_4 = 5$, $c_3 = 4$ c_i $(i = 2,3,4)$, c_j $(j = 1,2)$, c_k $(k = 2,3)$.

References

[1] S.Nicosia and P.Tomei, "Robot Control by Using Only Joint Position Measurements", IEEE Transactions on Automatic Control, pp1058-1061, 1990

[2] C.Canudas de Wit, N.Fixot, and K.Astrom, "Trajectory Tracking in Robot Manipulator via Nonlinear Estimated State Feedback", IEEE Transactions on R&A, pp138-144, 1992

[3] J.Yuan and Y.Stepanenko, "Robust Control of Robotic Manipulators Without Velocity Feedback", Int. Journal of Robust and Nonlinear Control, pp203-213, 1991

[4] H.Hajjir and H.M.Schwartz, "An Adaptive Nonlinear Output Feedback Controller for Robot Manipulators", Proceedings of the American Control Conference, pp. 1520-1524, 1999

[5] E.Zergeroglu, W.Dixon, D.Haste, and D.Dawson, "A Composite Adaptive Output Feedback Tracking Controller for Robotic Manipulator", Proceedings of the American Control Conference, pp. 3013-3017, 1999

[6] K.W.Lee and H.K.Khalil, "Adaptive Output Feedback Control of Robot Manipulators Using High-gain Observer", Int.J.Control, pp869-886, 1997

[7]F.Zhang, D.M.Dawson, M.S.de Queiroz and W.Dixon,"Global Adaptive Output Feedback Tracking Control of Robot Manipulator", Proceedings of IEEE the 36th conference on Decision and Control, pp3634-3639, 1997

[8] C.Canudas De Wit and N.Fixot,"Adaptive Control of Robot Manipulator via Velocity Estimated Feedback", IEEE Transactions on Automatic Control, pp1234-1237, 1992

[9] R.Colbaugh and K.Glass, "Adaptive Tracking Control of Manipulators Using Only Position Measurements", CDC, pp51-54, 1995

[10]R.Colbaugh and K.Glass, "Adaptive Regulation of Manipulators Using Only Position Measurements", Journal of Robotics Research, pp703-713, 1997

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

An optimal control model of a neuromuscular system in human arm movements and its control characteristics

○Tadashi KASHIMA [1], Yoshihisa ISURUGI [2], and Masasuke SHIMA [3]

[1] Kobe City College of Technology, Gakuen Higashimachi 8-3,Nishi-ku,Kobe,kashima@kobe-kosen.ac.jp
[2] Hokkaido University, North13,West8, Sapporo, isurugi@coin.eng.hokudai.ac.jp
[3] Hokkaido University, North13,West8, Sapporo, shima@coin.eng.hokudai.ac.jp

Abstract The joint torque to set human limbs into motion is generated by a group of muscles crossing the joint. As the activation of each muscle, which corresponds to the isometric force, is determined by a neural input, a neuromuscular system controlling all muscles has to be considered for understanding human movements. In this study, an optimal control model of a neuromuscular system is investigated and its control characteristics are examined. First, the criterion characterizing the optimal control model is defined by a derivative of isometric force of individual muscles and optimal trajectories in human arm movements are produced under varying conditions of target position. These trajectories are compared with the experimentally observed ones, and the results that optimal trajectories well demonstrate human arm movements are presented. Second, the behavior of individual muscles in various movements is quantitatively examined by means of the simulation results, and control characteristics of human neuromuscular system is reasonably explained.

Keywords; neuromuscular system, human arm movement, optimal trajectory

1. Introduction

The dynamics of human limbs is determined by the joint torque. Accordingly, a control mechanism of human movements can be analyzed by evaluating the torque[1] or its derivative[2]. However, the control mechanism may not be thoroughly explained, because the activation of individual muscles to determine the torque cannot be specified from such angle.

The effective approach to investigating human movements is to examine the mechanical and physiological property of a muscle and to discuss human trajectories based on the behavior of individual muscles. Here, optimal control models of minimum energy[3][4] and minimum isometric torque derivative[5] are proposed. However, the dynamics of a muscle is neglected and the coactivation of antagonist muscles is not considered in these models.

The isometric force of a muscle representing the muscle activation is controlled by a neural input, and there is a dynamic relation between the neural input and the isometric force. Such dynamic time delay is not assumed short enough to neglect comparing with human movement duration. Hence, one of the essential subjects to understand human movements results in the argument of a neuromuscular system. The authors[6] have proposed a criterion defined by a derivative of the isometric force of individual muscles and discussed the control characteristics mainly for elbow joint movements. However, this optimal model is not satisfactorily investigated for human arm movements with multi-joints, and the behavior of individual muscles has not been examined in detail.

In this study, human arm trajectories in natural unrestricted reaching movements are observed under conditions that sufficient kinds of target positions are provided. Subsequently, the optimal control model under the criterion proposed in the previous study[6] is applied for a musculoskeletal system of a human arm, and optimal trajectories are produced. Here, initial conditions with respect to the isometric force of individual muscles are assumed on the data given in our previous experiments of an elbow joint[6]. These optimal and observed trajectories are comparatively analyzed, and the results that our optimal model reproduces human arm trajectories are confirmed. In addition to such analysis at joint torque level, control characteristics of human neuromuscular system is discussed upon the simulation results. Consequently, load-sharing effect among muscles as well as coactivation of antagonist muscles is verified by evaluating the isometric force of individual muscles.

2. Dynamic equation

The isometric force of a muscle is dynamically coupled with the neural input, and the dynamics can be approximated by a first order differential equation[7]. In addition, time constant can be also estimated to be within the range 40-100 ms[6]. Consequently, the first order approximation with the time constant of 70 ms

as an averaged value is provided for the dynamics, and the dynamics can be represented as

$$\dot{f} = (1/\tau)(u - f), \qquad (1)$$

where f, τ, and u are isometric force, time constant, and neural input, respectively. After a muscle is activated by the input, a muscle delivers the output force to the limb. The output force is the function of the isometric force and the contracting velocity, and can be written for shortening ($v / v_{max} \geq 0$)

$$p = f(1 - v/v_{max})/(1 + 4v/v_{max}) \qquad (3)$$

and for lengthening ($-1/3.6 < v/v_{max} < 0$)

$$p = f(1 - 1.07v/v_{max})/(1 - 4.24v/v_{max}) \qquad (4)$$

where, p, v, and v_{max} are output force, contracting velocity, maximum shortening velocity, respectively.

A human arm with two movable joints in a vertical plane is modeled as in Fig.1. Here, muscles in number from #1 to #4 are uniarticular and those of #5 and #6 are biarticular. In addition, system parameters are measured or estimated by data taken from published article[8] and given in Table 1. From (1), the relation between the neural input and the

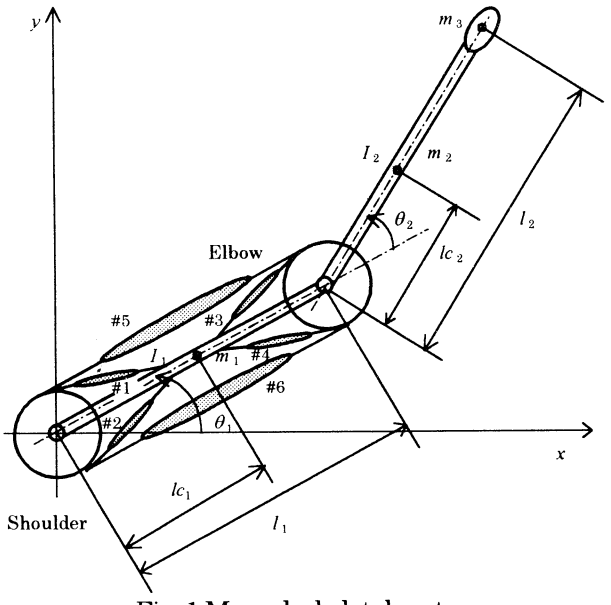


Fig. 1 Musculoskeletal system

Table 1 System parameters

Parameter	Value	Unit
M_1	1.8	Kg
M_2	1.1	Kg
M_3	0.4	Kg
L_1	0.28	M
l_{c1}	0.14	M
L_2	0.3	M
l_{c2}	0.1	M
I_1	0.013	Kgm^2
I_2	0.0065	Kgm^2

the isometric force of each muscle becomes

$$\dot{f}_i = (1/\tau)(u_i - f_i) \quad i = 1,2,\cdots,6 , \qquad (5)$$

where subscript i denotes muscle number in Fig.1. In addition, supposing that all muscles have the same moment arm l_m and the moment arm stays constant for any angle position, output force can be given by exchanging v/v_{max} for $\dot{\theta}/\dot{\theta}_{max}$ in (3) and (4). Here, maximum angular velocity θ_{max} is decided as 22 rad/s for flexors(#1, #3, and #5) and 28 rad/s for extensors(#2,,#4, and #6) with reference to the published data[9]. Consequently, the output force p becomes the function of isometric force and angular velocity, and the joint torque of the shoulder n_1 and the elbow n_2 can be expressed as

$$\begin{cases} n_1 = (p_1 + p_5 - p_2 - p_6) l_m \\ n_2 = (p_3 + p_5 - p_4 - p_6) l_m \end{cases} \qquad (6)$$

As the joint torque $\boldsymbol{n} = [n_1, n_2]^T$ is given by (6), the dynamic equation of an arm can be described as

$$\boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{n}(\dot{\boldsymbol{\theta}}, \boldsymbol{f}, \boldsymbol{u}), \qquad (7)$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2]^T$ is joint angle. Besides, $\boldsymbol{M}$, $\boldsymbol{h}$, and $\boldsymbol{g}$ in (7) are precisely discussed in our published paper[6]. Subsequently, we define the input vector $\boldsymbol{u}$ and the state vector $\boldsymbol{x}$ as

$$\boldsymbol{u} = [u_1, u_2, \cdots, u_6]^T \text{ and}$$

$$\boldsymbol{x} = [x_1, x_2, \cdots, x_{10}]^T = [\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2, f_1, f_2, \cdots, f_6]^T \qquad (8)$$

Then, the tenth-order state equation is given as

$$\dot{\boldsymbol{x}} = \boldsymbol{a} + \boldsymbol{B}\boldsymbol{u}$$

$$= \begin{bmatrix} \boldsymbol{x}_v \\ \boldsymbol{M}(\boldsymbol{x}_p)^{-1}(\boldsymbol{n}(\boldsymbol{x}_v, \boldsymbol{x}_f) - \boldsymbol{h}(\boldsymbol{x}_p, \boldsymbol{x}_v) - \boldsymbol{g}(\boldsymbol{x}_p)) \\ -\boldsymbol{f}/\tau \end{bmatrix} + \begin{bmatrix} \boldsymbol{B}_1 \\ \boldsymbol{B}_2 \\ \boldsymbol{B}_3 \end{bmatrix} \boldsymbol{u} \qquad (9)$$

where $\boldsymbol{x}_p = [x_1, x_2]^T$, $\boldsymbol{x}_v = [x_3, x_4]^T$ and $\boldsymbol{x}_f = [x_5, \cdots, x_{10}]^T$. In addition, $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$ are 2x6 matrix with all elements of 0 and $\boldsymbol{B}_3$ is 6x6 scalar matrix with the diagonal element of $1/\tau$.

3. Human arm trajectory

3.1 Experiment

Natural unrestricted reaching movements in a vertical plane were measured. Experiments were conducted and measured data was processed according to the method described in our previous paper[4]. Moreover, in this study, additional target position was provided in order to evaluate sufficient kinds of human movements. The arranged target positions cover the movement of the vertical, horizontal, and diagonal direction and of the large travel distance.

3.2 Simulation

The criterion to reduce the metabolic energy and also to promise the smooth trajectory is defined with the derivative of the isometric force[5][6] as

$$J_o = \frac{1}{2}\int_0^{t_f}\sum_{i=1}^{6}\dot{f}_i^{\,2}\,dt \quad . \tag{10}$$

As its derivative is given by (5), the criterion can be newly defined as

$$J = \frac{1}{2}\int_0^{t_f}\sum_{i=1}^{6}(u_i - f_i)^2\,dt = \frac{1}{2}\int_0^{t_f}\sum_{i=1}^{6}(u_i - x_{i+4})^2\,dt\,, \tag{11}$$

where t_f is movement duration. The optimal input to minimize the criterion of (11) is

$$u_i^{\circ} = \left\{u_i : \min_{u_i} H(\boldsymbol{x},\boldsymbol{u},\boldsymbol{\lambda})\right\} \quad i = 1,\cdots,6\,, \tag{12}$$

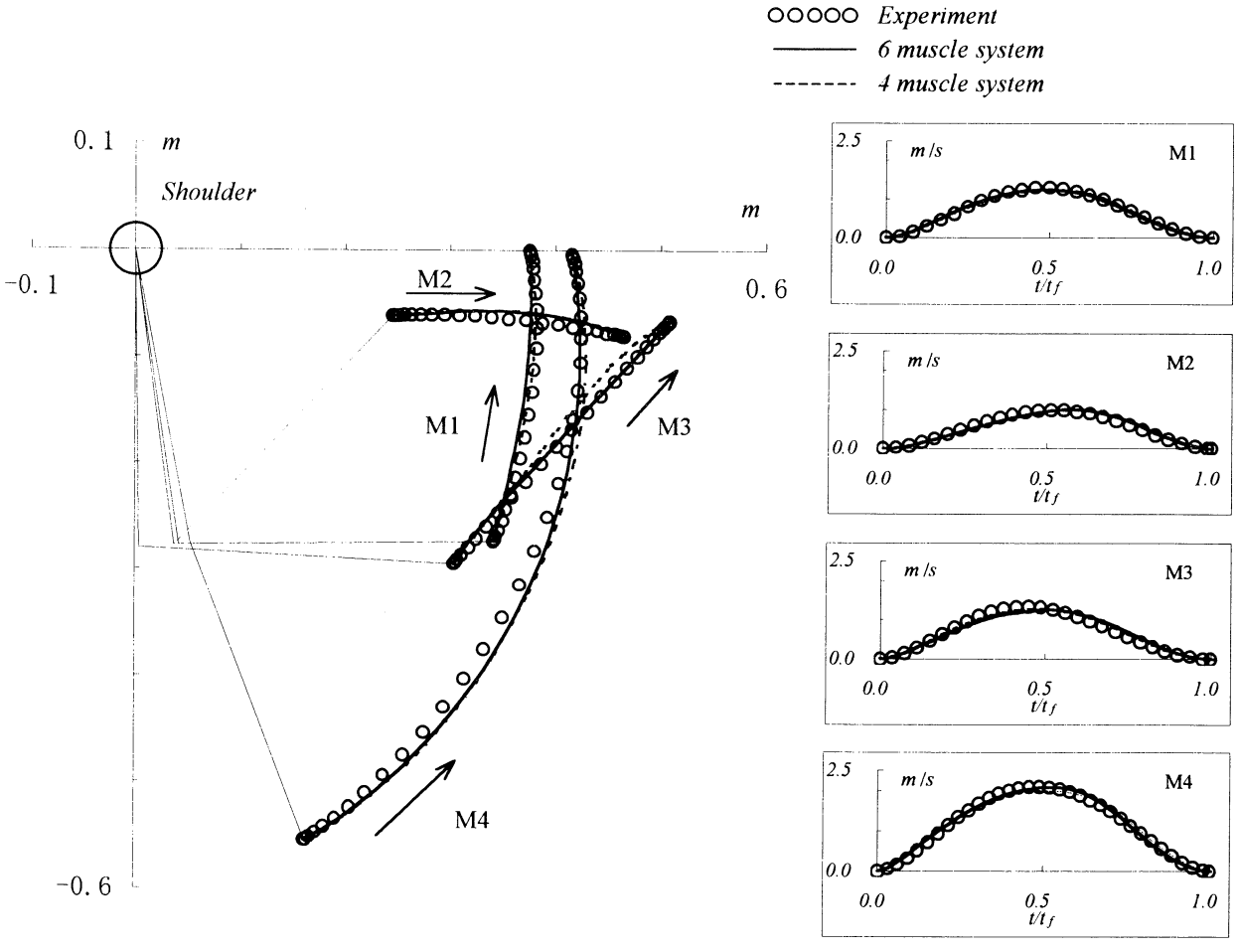
where $\boldsymbol{\lambda}$ is a 6x1 adjoint vector and H is a Hamiltonian defined by

$$H = \frac{1}{2}\sum_{i=1}^{6}(u_i - x_{i+4})^2 + \boldsymbol{\lambda}^T(\boldsymbol{a} + \boldsymbol{B}\boldsymbol{u}) \tag{13}$$

Here, a set of necessary conditions to have a stationary value is

$$\dot{\boldsymbol{x}} = \partial H(\boldsymbol{x},\boldsymbol{u}^{\circ},\boldsymbol{\lambda})/\partial\boldsymbol{\lambda}\,,\quad \dot{\boldsymbol{\lambda}}^T = -\partial H(\boldsymbol{x},\boldsymbol{u}^{\circ},\boldsymbol{\lambda})/\partial\boldsymbol{x} \tag{14}$$

These 2x10 differential equations can be solved if boundary conditions $\boldsymbol{x}(0)$ and $\boldsymbol{x}(t_f)$ are prepared. Hereupon, $x_i(0)$ and $x_i(t_f)$ for i=1,2 are given by the experimental results and $x_i(0)$ and $x_i(t_f)$ for i=3,4 are set as 0. Nevertheless, it is an essential issue to suppose the isometric force of individual muscles $x_i(0)$ and $x_i(t_f)$ for i=5,$\cdots$,10. The authors have measured EMG(electromyogram) of a flexor and an extensor crossing an elbow joint and have estimated the isometric force of these muscles in a steady state condition[6]. In that experiment, the isometric force of an extensor was observed in a specific rate when a flexor was activated, and vice versa. According to the observation, the activation rate for an antagonistic muscle can be estimated about 10 % at the beginning and the end of motion. In addition to this result, the following conditions are provided to determine the isometric force of individual muscles: 1)the angular acceleration $\ddot{\boldsymbol{\theta}}(0) = \ddot{\boldsymbol{\theta}}(t_f) = \boldsymbol{0}$ and 2) square sum of isometric force is minimized. Thus, all boundary conditions are determined and the optimal trajectory can be produced in accordance with an improved transition matrix algorithm[6].



(a) Hand path

(b) Hand velocity

Fig.2 Human arm trajectory

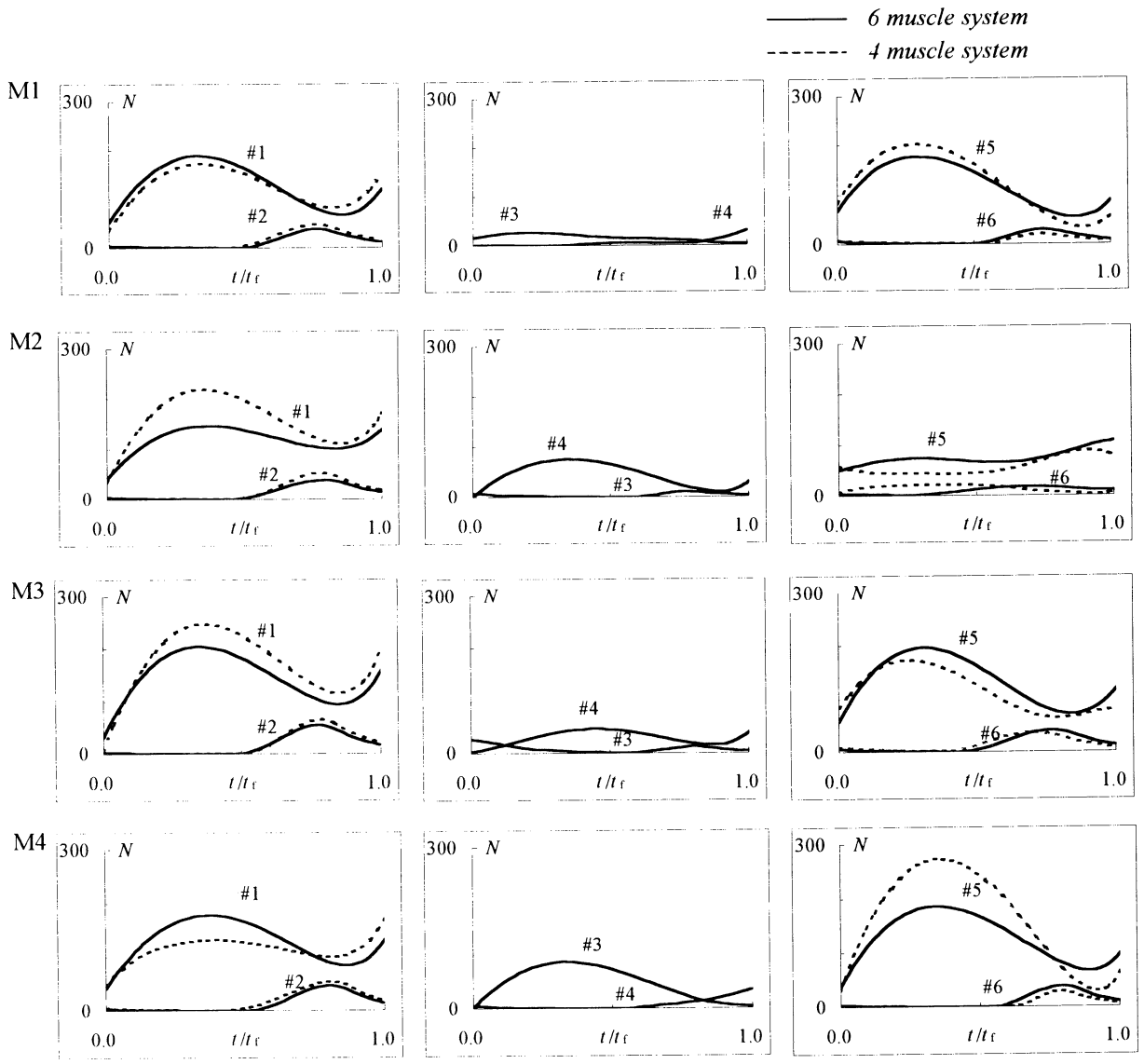


Fig. 3 Isometric force of muscle

4. Results and Discussions

The hand paths and velocities for the observed trajectories are shown in Fig.2. The hand paths of M1 ,M2 and M3 are nearly straight. In contrast, the hand path of M4 is obviously curved and a full explanation for why certain movements are curved is not obtained from kinematic approach such as minimum jerk hypothesis[10].

The optimal trajectories shown in Fig.2 are produced under 2 kind of musculoskeletal systems. One is the basic system with 6 muscle arrangement (named 6 muscle system) explained in Fig.1 and the other is the system with 4 muscle arrangement (named 4 muscle system), in which #1 and #2 muscles are eliminated. As can be seen from the Figure, the hand paths and velocity profiles of 6 muscle system are remarkably similar to experimental results. Especially, the strong curved path of M4 observed in the experiment is well demonstrated by the produced trajectory, and this is the distinguishing feature of this optimal model. As regards the trajectories of 4 muscle system, all hand paths are very similar to those of 6 muscle system and hand velocity profiles of the two systems are totally overlapped. This result shows that the basic characteristics of human trajectories at the joint torque level can be examined with 4 muscle system.

In addition to such analysis with respect to the hand paths and velocity profiles, the behavior of individual muscles has to be examined for further understanding of human movements. However, respective isometric force for all muscles crossing a single joint may not be experimentally measured at this moment. Accordingly, the behavior of the individual muscles is discussed on the basis of the simulation results shown in Fig.3. First of all, we note

that all muscles including 3 sets of a flexor and an extensor are not necessary to drive a human arm. For example, any movement can be achieved without the uniarticular muscles(#3 and #4), and this is the case of 4 muscle system. As regards the isometric force of 4 muscle system, flexors(#1 and #5) are mainly activated in all movements. This is the reasonable result, because these movements mostly require positive torque for all joints. For 6 muscle system, the same flexors are also activated but an extensor(#4) indicates higher level of isometric force than a flexor(#3) in the movements of M2 and M3. In addition, load-sharing effect among all sets of a flexor and an extensor can be appreciated especially for the movements of M2 and M4. Table 2 is provided to quantitatively investigate these phenomena.

Table 2 Performance criteria values

Motion	System	J^{*1}	J_{ref}^{*2}
M1	6 muscle	1,621	7,131
	4 muscle	1,751	7,414
M2	6 muscle	676	4,920
	4 muscle	1,321	6,665
M3	6 muscle	2,126	10,035
	4 muscle	2,483	11,004
M4	6 muscle	1,485	11,037
	4 muscle	2,144	13,211

$$*1;\ J=\frac{1}{2}\int_0^{t_f}\sum_i (u_i - x_{i+4})^2\,dt\ ,\ *2;\ J_{ref}=\frac{1}{2}\int_0^{t_f}\sum_i x_i^{\,2}\,dt$$

Here, the value of criterion(J^{*1}) of 6 muscle system, which indicates the change of muscle activation, shows 30% to 50 % smaller values than that of 4 muscle system in movements except M1. Besides, reduction of 5% to 25% for criterion(J_{ref}^{*2}), which exhibits muscle activation, is recognized for 6 muscle system. Consequently, load among muscle sets is effectively shared in 6 muscle system with muscle redundancy, and such redundancy contributes to the decrease of muscle activation and its change.

In addition to such behavior, Fig. 3 shows an interesting feature that every antagonistic muscle activation of all muscle sets is evidently increased near the end of motion. As such coactivation of antagonist muscles induces the increase of joint impedance, this can be explained as a reasonable phenomenon in a deceleration process of the movement.

5. Conclusion

In this study, an optimal control model of a neuromuscular system has been precisely investigated for human arm movements, and its control characteristics have been analyzed. As regards the human arm trajectories, it is verified that our optimal control model produces the similar joint torque profile as human does. Moreover, the behavior of individual muscles has been quantitatively analyzed under two types of musculoskeletal systems. As a result, load-sharing effect among muscle sets is recognized for 6 muscle system in most movements and the advantages of muscle redundancy of a musculoskeletal system in a human arm is confirmed. Besides, simultaneous activation of a flexor and an extensor is seen near the end of movement, and this phenomenon encourages the increase of joint impedance. However, these theoretical analyses with respect to individual muscles have to be verified by experiments, and this is the issue to be discussed in the future.

References

[1] Kashima T, Isurugi Y(1995), Trajectory formation in human arm movements(in Japanese). Trans of SICE 34-10:1416-1422

[2] Uno Y, Kawato M, Suzuki R(1989), Formation and Control of Optimal Trajectory in Human Multi-Joint Arm Movement. Biol.Cybern.61:89-101

[3] Alexander R(1997), A minimum energy cost hypothesis for human arm Trajectories. Biol. Cybern. 76:97-105

[4] Kashima T, Isurugi Y(1998), Human arm trajectory formation (in Japanese). Trans of SICE 34-10:1416-1422

[5] Kashima T, Isurugi Y(1997), Trajectory formation based on physiological characteristics of skeletal muscles. Biol. Cybern. 78:413-422

[6] Kashima T, Isurugi Y, Shima M(2000), Analysis of a muscular control system in human movements. Biol. Cybern. 82:123-131

[7] Hannaford B(1990), A nonlinear model of the phasic dynamics of Muscle Activation. IEEE Trans. Biomed. Eng. 37:1067-1075

[8] US Department of Commerce NTIS(1975), Investigation of inertial properties of the human body. AD-A016 485:72-80

[9] Winters JM, Stark L(1985), Analysis of fundamental human movement patterns through the use of in-depth antagonistic muscle models. IEEE Trans. Biomed. Eng. 32:826-839

[10] Atkeson C, Hollerbach J(1985), Kinematic features of unrestrained vertical arm movements. J. Neurosci. 5:2318-2330

[11] Flash T, Hogan N(1985), The coordination of arm movements :an experimentally confirmed mathematical model, J. Neurosci. 5:1688-1703

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A parametrization of all linear observer for the minimum phase systems

Kou Yamada
Department of mechanical system engineering
Gunma University
Kiryu, 376-8515 Japan

Abstract

In the present paper, an exact parametrization is given for the class of all linear observers for linear minimum phase systems. At first, the parametrization of all linear observer for the minimum phase systems that is described in the transfer functional description is given. A parametrization of all linear observer with unknown input and that of all linear observer with unknown output are clarified. A relation between linear observer for the nonminimum phase system and the minimum phase system are shown. Finally, the state space description of all linear observer for the minimum phase systems that is described in the state space description is given.

1 Introduction

In the present paper, we examine a parametrization for the class of all linear observers for multiple-input/multiple-output time-invariant continuous-time minimum phase systems. The parametrization problem is the problem in which all stabilizing controllers for a plant or in which all observer for a plant are sought [1]-[9]. The parametrization is used in several control problems for example H_∞ control [10] and strong stabilizing control problem [11].

The parametrization problem to find all stabilizing controllers for the plant were considered by [1]-[6]. For an unstable plant, the structure of a parametrization of all stabilizing controllers for the plant has full-order state feedback, including a full-order observer [3]. For an asymptotically stable plant, a parametrization of all stabilizing controllers for the plant has a structure identical to that of Internal Model Control [5]. Since this parametrization can successfully search for all proper stabilizing controllers, it is used as a tool for many control problems. According to [3, 5], the difference between control structure of stabilizing controller for the unstable plant and that of asymptotically stable plant was clarified. Glaria and Goodwin [6] gave a simple parametrization of all stabilizing controllers for the minimum phase systems; however, one practical difficulty remains: the parametrization of the stabilizing controller given by Glaria and Goodwin generally includes improper controllers. In practical application, the controller $C(s)$ is required to be proper. If a parametrization of all proper stabilizing controllers for the plant can be obtained, a parametrization of all stabilizing controllers for minimum phase systems can be applied to several applications. Yamada expand the result by Glaria and Goodwin and gave exact parametrization of all proper stabilizing controller for the minimum phase systems [8]. In this way, the parametrization of all stabilizing controller for the plant was completely solved.

On the other hand, the parametrization of all observers for the linear systems was few studied. Ding, Guo and Frank gave the parametrization of all linear observer for the system was considered [9]. Ding, Guo and Frank also considered the relationship between these parametrization and the Luenberger type observer.

In the present paper, we give simple parametrization of all proper linear observers for the multiple-input/multiple-output continuous time minimum phase systems. At first, the parametrization of all linear observer for the minimum phase systems that is described in the transfer functional description is given. A parametrization of all linear observer with unknown input and that of all linear observer with unknown output are clarified. There exists no freedom of both type of observer is shown, that is, there exists only one linear observer with unknown input and that with unknown output, each other. A relation between linear observer for the nonminimum phase system and the minimum phase system are shown. In addition, the state space description of all linear observers for the minimum phase systems that is described in the state space description is given.

2 Problem formulation

Let us consider a multiple-input/multiple-output continuous time system $G(s) \in R^{m\times m}(s)$. The state space description of $G(s)$ is denoted by

$$\begin{cases} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{cases} \tag{1}$$

Here $A \in R^{n\times n}$, $B \in R^{n\times m}$, $C \in R^{m\times n}$ and $D \in R^{m\times m}$, $x \in R^n$ is the state variable, $u \in R^m$ is the control input, $y \in R^m$ is the output and (C, A) is observable. Here, $G(s)$ is assumed to be proper and of minimum phase, that is

$$\text{rank} \begin{bmatrix} A - sI & B \\ C & D \end{bmatrix} = n + m \ \ \forall \Re\{s\} \geq 0, \ s \in C. \tag{2}$$

In addition $G(s)$ is assumed to be invertible, that is

$$\text{rank } G(s) = m. \tag{3}$$

In the present paper, we seek a parametrization of all linear observer $\hat{r}$ for $r(t)$ denoted by

$$r(t) = Px(t), \tag{4}$$

where $P \in R^{k\times n}$ is given matrix. Here we call $\hat{r}$ a observer for r if

$$\lim_{t\to\infty} (r(t) - \hat{r}(t)) = 0 \tag{5}$$

for all $u(t)$ and any initial condition [9].

The available measurement for the observer design are the input $u(t)$ and the output $y(t)$. Therefore the general form of a linear observer is written by

$$\hat{r}(s) = F(s)u(s) + H(s)y(s) \in R^k(s), \tag{6}$$

where $F(s)$ and $H(s)$ are asymptotically stable transfer function. Therefore, in the present paper, we give exact parametrization of all $F(s)$ and $H(s)$ in (6) to satisfy (5).

3 Parametrization

In the present section, the parametrization of all linear observer $\hat{r}$ for r is given using transfer functional description.

The parametrization of all linear observer for r is summarized following theorem.

Theorem 1 *$G(s)$ is assumed to be of minimum phase. $\hat{r}(s)$ in (6) is linear observer for r if and only if $F(s)$ and $H(s)$ in (6) are written by*

$$F(s) = -Q(s) \tag{7}$$

and

$$H(s) = P(s) + Q(s)G_m^{-1}(s) \tag{8}$$

respectively. Here $P(s)$ is written by

$$P(s) = P(sI - A)^{-1}B \tag{9}$$

and $Q(s) \in R^{k\times m}(s)$ is any asymptotically stable rational function such that $Q(s)G_m^{-1}(s)$ is proper.

Proof of this theorem requires following theorem.

Theorem 2 *$\hat{r}(s)$ in (6) is linear observer for r if and only if $F(s)$ and $H(s)$ in (6) are written by*

$$F(s) = P(s)Y_u(s) - Q(s)\hat{N}_u(s) \tag{10}$$

and

$$H(s) = P(s)X_u(s) + Q(s)\hat{M}_u(s) \tag{11}$$

respectively [9]. Here $\hat{N}_u(s)$ and $\hat{M}_u(s)$ are coprime factor of $G(s)$ written by

$$G(s) = \hat{M}_u^{-1}(s)\hat{N}_u(s), \tag{12}$$

$Q(s) \in R^{k\times m}(s)$ is any asymptotically stable rational function, $Y_u(s)$ and $X_u(s)$ are asymptotically stable rational function satisfying Bezout identity

$$\begin{aligned} &\begin{bmatrix} Y(s) & X(s) \\ -\hat{N}(s) & \hat{M}(s) \end{bmatrix} \begin{bmatrix} M(s) & -\hat{X}(s) \\ N(s) & \hat{Y}(s) \end{bmatrix} \\ &= \begin{bmatrix} M(s) & \hat{X}(s) \\ -N(s) & \hat{Y}(s) \end{bmatrix} \begin{bmatrix} Y(s) & -X(s) \\ \hat{N}(s) & \hat{M}(s) \end{bmatrix} \\ &= \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}. \end{aligned} \tag{13}$$

and $Q(s) \in R^{k\times m}(s)$ is any asymptotically stable rational function.

Theorem 1 is proved using the above described theorem.

(Proof) From the assumption that $G(s)$ is of minimum phase, one of the coprime factorization of $G(s) = N(s)M^{-1}(s) = \hat{M}^{-1}(s)\hat{N}(s)$ are given by

$$N_u(s) = \hat{N}_u(s) = I \tag{14}$$

and

$$M_u(s) = \hat{M}_u(s) = G^{-1}(s). \tag{15}$$

When let $N_u(s)$ and $\hat{N}_u(s)$ (14) and $M_u(s)$ and $\hat{M}_u(s)$ be defined by (15), a solution of Bezout equation (13) is given by

$$X_u(s) = I \tag{16}$$

$$Y_u(s) = 0. \tag{17}$$

From Theorem 2, a parametrization of all linear observer for r is given by

$$\hat{r}(s) = F(s)u(s) + H(s)y(s) \tag{18}$$

$$F(s) = -Q(s) \tag{19}$$

$$H(s) = P(s) + Q(s)G^{-1}(s). \tag{20}$$

We have thus complete the proof of Theorem 1. ∎

Note 1 *In practical application, both $F(s)$ and $H(s)$ are required to be proper. The condition that both $F(s)$ and $H(s)$ are proper is that $Q(s)$ and $Q(s)G^{-1}(s)$ are proper.*

From Theorem 1, a parametrization of all linear observer with unknown input, that is a linear observer without using the control input $u(s)$, is obtained as follow.

Corollary 1 *$G(s)$ is assumed to be of minimum phase. $\hat{r}(s)$ in (6) is linear observer with unknown input for r, that is $\hat{r}(s) = H(s)y$ is linear observer for r if and only if $H(s)$ in (6) are written by*

$$H(s) = P(s). \tag{21}$$

(Proof is omitted)

In addition, from Theorem 1, there exists linear observer with unknown output if the minimum realization of $P(s)G(s)$ is asymptotically stable. The parametrization of all linear observers with unknown output is summarized as following corollary.

Corollary 2 *$G(s)$ is assumed to be of minimum phase. There exists a linear observer with unknown output for $r(s)$, that is $\hat{r}(s) = F(s)u$ is linear observer for r if and only if the minimum realization of $P(s)G(s)$ is asymptotically stable. When the minimum realization of $P(s)G(s)$ is asymptotically stable, $\hat{r}(s)$ in (6) is linear observer with unknown output for r, that is $\hat{r}(s) = F(s)u$ is linear observer for r if and only if $F(s)$ in (6) are written by*

$$F(s) = \bar{P}(s), \tag{22}$$

where $\bar{P}(s)$ is the minimum realization of $P(s)G(s)$.

(Proof is omitted)

From Theorem 1, a parametrization of all linear observer $\hat{r}$ for r is rewritten by

$$\begin{aligned}\hat{r}(s) &\\ &= -Q(s)u(s) + \left\{P(s) + Q(s)^{-1}(s)\right\} y(s) \\ &= P(s)G(s)u(s) + \left\{P(s) + Q(s)G^{-1}(s)\right\} \\ &\quad \{y(s) - G(s)u\}. \end{aligned} \tag{23}$$

The second term of above equation works only there exists the disturbance and the initial state of x is not equal to 0. Therefore $Q(s)$ is settled to specify the characteristics of linear observer for the disturbance and the initial state. Main difference between the linear observer for the nonminimum phase system and the minimum phase system is that when the disturbance does not exists and the initial state is equal to zero matrix. When the disturbance does not exists and the initial state is equal to zero matrix, the linear observer for the minimum phase system can estimate the state variable x using the output y because of $G(s)u(s) = y(s)$ and (23). But that for the nonminimum phase system cannot estimate the state variable x only using the output y.

4 Free parameter $Q(s)$

In the previous section, a parametrization of all linear observer for r is given and it is shown that the free parameter $Q(s)$ and $Q(s)G^{-1}(s)$ must be proper. In the present section, a parametrization of $Q(s)$ such that both $Q(s)$ and $Q(s)G^{-1}(s)$ is proper is given.

A parametrization of all asymptotically stable $Q(s)$ such that both $Q(s)$ and $Q(s)G^{-1}(s)$ is proper are summarized as follow.

Theorem 3 *$Q(s)$ is assumed to be asymptotically stable and proper and $G(s)$ is assumed to be of minimum phase. In addition, the poles of $Q(s)$ is not equivalent to that of $G(s)$. Then $Q(s)$ and $Q(s)G^{-1}(s)$ are proper if and only if $Q(s)$ is written by*

$$\begin{aligned} Q(s) &\\ &= \left[\begin{array}{c|c} T_{11}^{-1}\hat{A}T_{11} & T_{11}^{-1}\hat{B}T_{11}D - T_{11}^{-1}T_{12}T_{22}^{-1}B \\ \hline \hat{C}T_{11} & \hat{D}D \end{array}\right], \end{aligned} \tag{24}$$

where T_{11} and T_{22} are any nonsingular matrix with appropriate size, $\hat{A} \in R^{p\times p}$ is any matrix that has no poles in the closed right half plane, T_{12}, $\hat{B}$, $\hat{C}$ and $\hat{D}$ are any matrices with appropriate size that satisfy

$$T_{12}T_{22}^{-1}AT_{12} - \hat{A}T_{12} - \hat{B}CT_{22} = 0 \tag{25}$$

and

$$\hat{C}T_{12} + \hat{D}CT_{22} = 0. \tag{26}$$

(Proof is omitted)

To design $Q(s)$, we must find T_{11}, T_{12}, T_{22}, $\hat{A}$, $\hat{B}$, $\hat{C}$ and $\hat{D}$ to satisfy Theorem 3.

From the proof of Theorem 3, T_{11}, T_{12} and T_{22} works only to similarly transformation matrix. This implies that the transfer function of $Q(s)$ is independent of the selection of T_{11}, T_{12} and T_{22}. In addition, it is obvious that any proper $Q(s)$ can be factorized by

$$Q(s) = \bar{Q}(s)Q_0(s), \tag{27}$$

where $\bar{Q}(s)$ is proper asymptotically stable rational function matrix and $Q_0(s)$ is biproper asymptotically stable rational function matrix such that $\lim_{\omega\to\infty}\{Q_0(j\omega)G^{-1}(j\omega)\} = I$. From these facts, we have following theorem.

Theorem 4 *$Q(s)$ is assumed to be asymptotically stable and proper and $G(s)$ is assumed to be of minimum phase. In addition, the poles of $Q(s)$ is not equivalent to that of $G(s)$ and the freedom of state space description of $Q(s)$ are removed. Then $Q(s)$ and $Q(s)G^{-1}(s)$ are proper if and only if $Q(s)$ is written by*

$$Q(s) = \bar{Q}(s)\left[\begin{array}{c|c} A-\hat{B}C & \hat{B}D-B \\ \hline -C & D \end{array}\right], \tag{28}$$

where $\hat{B}$ is any matrix with appropriate size such that all of eigenvalue of $A-\hat{B}C$ are in the open left half plane.

(Proof is omitted)

5 Conclusion

In the present paper, we gave exact parametrization of all proper observers for the multiple-input/multiple-output continuous time minimum phase systems. At first, the parametrization of all linear observer for the minimum phase systems that is described in the transfer functional description was given. The state space description of all linear observer for the minimum phase systems that is described in the state space description was given.

Acknowledgements

The author would like to express his gratitude to Prof. K. Watanabe of Yamagata University, Prof. M. Ikeda of Osaka University and Prof. Y. Ohta of Osaka University for their helpful discussions. This research was partially supported by Takayanagi Foundation for electronics science and technology.

References

[1] G. Zames, "Feedback and optimal sensitivity: model reference transformations, multiplicative seminorms and approximate inverse," *IEEE Trans. on AC*, AC26, pp.301-320, 1981.

[2] D.C. Youla, H. Jabr and J.J. Bongiorno, "Modern Wiener-Hopf design of optimal controllers. Part I," *IEEE Trans. on AC*, AC21, pp. 3-13, 1976.

[3] K. Zhou, J.C. Doyle and K. Glover, "Robust and optimal control," *Prentice-Hall*, 1996.

[4] C.A. Desoer, R.W. Liu, J. Murray and R. Saeks, "Feedback system design: The fractional representation approach to analysis and synthesis," *IEEE Trans. on AC*, AC25, pp.399-412, 1980.

[5] M. Morari and E. Zafiriou, "Robust Process Control," *Prentice-Hall*, 1989.

[6] J.J. Glaria and G.C. Goodwin, "A parametrization for the class of all stabilizing controllers for linear minimum phase systems," *IEEE Trans. on AC*, AC39-2, pp.433-434, 1994.

[7] M. Vidyasagar, "Control System Synthesis–A factorization approach–," *MIT Press*, 1985.

[8] K. Yamada, "A parameterization of all stabilizing controllers for linear minimum phase single-input and single-output systems," *Proc. IEEE Conf. Systems Man and Cybernetics'99*, Vol.V, pp.68-73, 1999.

[9] X.Ding,L.Guo and P.M.Frank, "Parametrization of linear observers andits application to observer design," *IEEE Trans. on AC*, AC39-8, pp.1648-1652, 1994.

[10] J.C.Doyle, K. Glover, P.P.Khargonekar and F.A.Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic Control*, AC-34, pp.831-847, 1989.

[11] G. Obinata and J.B.Moore, "Characterization of controllers in sumultaneous stabilization," *Systems & Control Letters*, 10, pp.331-340,1988.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Motion Intelligence to Use Dynamical Interfarances and Nonliner Friction of Mobile Manipulators

Mamoru MINAMI, Atsushi TAMAMURA, Toshiyuki ASAKURA, Fukui University

Abstract: In this research, we propose a concept of motion intelligence of a mobile manipulator which does not have driving power. The motion intelligence is defined here as that the mobile manipulator can find by itself a way to use its dynamical coupling and nonlinear friction for an objective motion, that is here, traveling motion. We confirmed by real experiments that mobile manipulator could find an effective motion of an arm mounted on a vehicle which makes it travel forward.

Key Words: Mobile Manipulator,Motion Intelligence,Fourier Series,Genetic Algorithm

1 Introduction

Mobile manipulators being able to move and work have been studied recently. A mobile manipulator consists generally of a vehicle to move and a mounted manipulator to work, like as a human that possesses legs and arms. The dynamical interferences of the mobile manipulator have been thought that they should be erased to improve control accuracy of moving and working operation. Humans are able to run faster by using their arm's swinging motion, that means a human is using the dynamical interferences of arms to body for the purpose to run faster. This action that humans improve arm's swinging motion for running can be thought as a motion intelligence, which is defined here as the ability to use indirectly related motion to the objective motion. We think that the mobile manipulator can get the motion intelligence if it can find a method and possess such function by itself to realize the given objective motion.

In this presentation, we aimed to give a mobile manipulator the motion intelligence that makes use of dynamical interferences of arm's swinging motion, which have not been used so far in the improvement of a traveling motion of the mobile manipulator. We propose a method to give motion intelligence, which evaluates by GA the motion trajectory that resulted by desired trajectory of the arm using Fourier Series and PD controller, and we show the result of experiments and simulation by which we confirmed that the mobile manipulator can travel without driving units, that means it possessed motion intelligence.

2 1 link mobile manipulator

Figure 1 shows 1-link mobile manipulator used in this research. The vehicle's wheel angle q_0 and the mounted-arm's angle q_1 are set vertical direction to be 0, and clockwise to be positive. The mounted arm is driven by a velocity-control-type amplifier and a servo motor possessing one dimensional freedom. The vehicle does not have driving motors nor breaks, the angle of wheel is simply observed by an encoder set beneath the center of the mobile manipulator. The vehicle's traveling is induced by the motion of arm driven by a PD controller and desired arm's motion $q_{1d}(t)$. From previously examined simulation studying, we confirmed that nonlinear friction plays an important role to make travel the mobile manipulator one way. Without the nonlinear friction, that in static friction, the mobile manipulator move merely oscillationally without one way traveling. Then, we are interested in a question that the machine can find a method for one way traveling by using the dynamical coupling and nonlinear friction, which are the inherent dynamics of mobile manipulator. Our proposal to put the mobile manipulator the motion intelligence is that the traveling motions induced by arm's motion made by PD controller and desired arm's motion defined with Fourier Series, are evaluated by GA possessing genes in which the coefficients of Fourier Series are included.

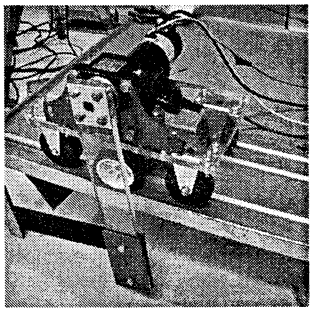


Fig. 1: 1-link mobile manipulator

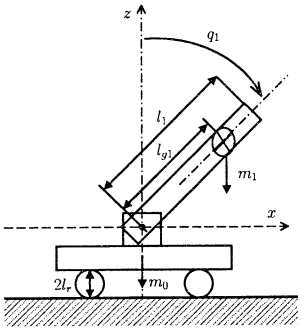


Fig. 2: model

3 Evolution of Motion

How to evolve the traveling motion is discussed here. A desired cyclic motion function of the mounted arm $q_{1d}(t)$ is expressed by using Fourier Series.

$$q_{1d}(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty}(a_n \cos(\frac{2n\pi}{T}t) + b_n \sin(\frac{2n\pi}{T}t)) \quad (1)$$

Where, $q_{1d}(t)$ is a function with period T, $a_0/2$ is the average value, $a_n, b_n (n = 1 \sim \infty)$ represent n-th Fourier coefficient. Being impossible to calculate the summation to infinite, the desired function is made of limited m terms. The coefficients $T, a_0, a_1, b_1, \cdots, a_m, b_m$ of the number $(2 + 2m)$ are coded in genes with population p in GA. The j-th gene in the i-th generation $\boldsymbol{g}_j^i$ is coded like as,

$$\boldsymbol{g}_j^i = [T_j^i, {a_0}_j^i, {a_1}_j^i, {b_1}_j^i, \cdots, {a_m}_j^i, {b_m}_j^i] \quad (2)$$

It is necessary for GA to give some judging standard to evolve the population toward the desired motion, that is here traveling. In GA, the standard of judgment is defined as fitness, that is a matching degree of an specified animal to environment, and the direction of the evolution can be changed by the fitness. The purpose of the

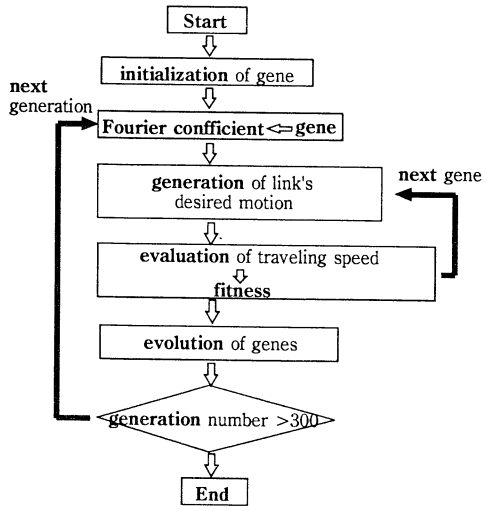


Fig. 3: flow chart

motion is to travel one way as fast as it can. Therefore the traveling average velocity is chosen as a fitness. Then Fourier coefficients being able to travel faster are to be selected in a sense of their combination and their values. The gene is decided by expression Eq.(2), and the Fourier Series give the desired motion $q_{1d}(t)$. The desired motion of the arm defined by Eq.(2) through $\boldsymbol{q_j^i}$, is evaluated by average traveling velocity during three-periods $(\boldsymbol{3T})$ motion of the arm, such as,

$$f_{1j}^{i} = \frac{1}{3T_j^i}(q_{0j}^{i}(3T_j^i) - q_{0j}^{i}(0)) \tag{3}$$

By evolving genes based on the traveling results, the best arm's motion being able to generate traveling motion could be found by itself through the dynamics and nonlinear friction.

4 Motion intelligence for plural purposes

So far we have discussed how to find the way to travel faster, in this section, we want to propose a method to purpose plural purposes. As a definite objective for a mobile manipulator, we added next object to the previous traveling objective, such as collision avoidance of the mounted arm with the floor. To make the GA search the plural objectives, we defined the following fitness function including the two objectives mentioned above.

$$f_p^i = f^i \times p \tag{4}$$

$$p = \begin{cases} 1 & \text{(without collision)} \\ \frac{1}{1+\alpha|I_1\dot{q}_{1imp}|} & \text{(with collision)} \end{cases} \tag{5}$$

where $\dot{q}_{1imp}$ means angular velocity at the moment of the clash, and I_1 does inertia moment of the arm, as the result, $I_1\dot{q}_{1imp}$ represents angular momentum at the time of impact. α is a weighting parameter to evaluate the collision. When the mounted arm collides with the floor, p has a value less than $\mathbf{1.0}$ as a penalty and reduce the fitness value depending on the strength of collision. Figure.3 shows the flow of

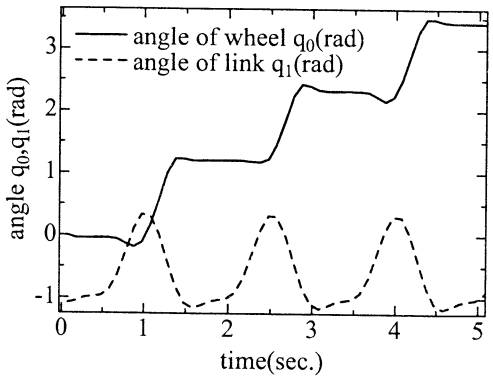


Fig. 4: Experimented travelling motion

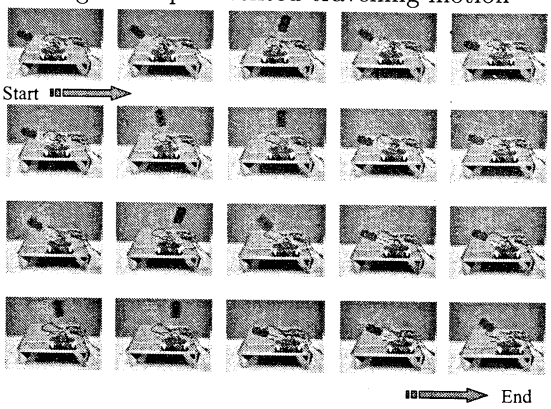


Fig. 5: confirmation by the experiment

making a desired function of the arm and of evaluating the resulted motion, and then improving according to the fitness function. In Fig.4, the obtained motion results to satisfy the plural objectives, which means that the fastest traveling motion without collision, are depicted. Chosen coefficient of Fourier Series by the GA are $T = \mathbf{1.5}, a_0 = \mathbf{-0.6}, a_1 = \mathbf{-0.35}, b_1 = \mathbf{-0.605}, a_2 = \mathbf{-0.17}, b_2 = \mathbf{0.16}$, and the average velocity is $\mathbf{2.4}(cm/s)$. The photographs of the motion recorded at every $\mathbf{0.3}$ second are shown in Fig.5. These results is representing that our method could find the effective motion of arm to travel one way without collision.

5 Conclusion

From this report, it has checked that mobile manipulator could obtain the motion intelligence by giving the system the ability to evaluate the effectiveness of the dynamical interferences and nonlinear friction without collision, and that it can pursue plural objectives.

6 References

1. Minami, Asakura, Huziwara, Kanbara, "A Proposal of Inverse Dynamics Compensation Method for PWS(omit-type) Mobile Manipulators", Transactions of The Japan Society of Mechanical Engineers (C), 1995, 61, 591, pp. 4351-4358

2. Minami, Asakura, Senda, Asakura, "The Travel of Mobile Manipulator using Dynamical Interference without Drive Unit", Lecture Transactions of 12th Advanty Symposium, 1999, pp.91-94

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A Hierarchical Approach for Environmental Adaptive Locomotion Control of Multi-legged Robot

Tadashi Odashima [1)] Zhi-wei Luo [2)1)] Shigeyuki Hosoe[3)1)]
1)Bio-Mimetic control research center RIKEN
Anagahora, Shimoshidami Moriyama-ku,Nagoya.463-0003
2)Faculty of Engineering Yamagata University
Jonan,Yonezawa,992-8501
3)Faculty of Engineering Nagoya University
Furocho,Chikusa-ku, Nagoya.464-8603

Abstract

This paper proposes a hierarchical control structure for a multi-legged robot to generate its gate with respect to the environmental constrains. The system consists of two layers, upper and lower. The upper layer receives the sensor information about the environment and plans the movement for the robot body. This plan is then sent to all subsystems at lower layer, and each subsystem determines its own motion with respect to this information as well as the local interaction between neighbor subsystems. We succeeded in implementing this approach to a real six-legged robot. Experimental results show the efficiency of this approach.

Key Words: Multi-legged Robot, Hierarchical Structure, Locomotion Pattern, Decentralized Cooperation.

1 Introduction

This paper studies the problem on how to control the locomotion of a Multi-legged robot in a complex environment, for example, walking along unknown curves. Generally, since the legged robot has many degrees of freedom, in order to realize the adaptive locomotion under the environmental constraints, enormous computations is needed for motion planning and controlling the trajectory of all legs, which makes it difficult for the real time control and adaptation. One answer to this problem is found in animals. Animals can flexibly change their locomotion pattern with respect to their environment and aims. This is realized in the following two levels. The configuration of the body with respect to the environment is planned at the brain level. Meanwhile, the gait patterns are autonomously generated at a lower level of nervous system. The lower nervous system called CPG (Central Pattern Generator) consists of many oscillators [1]. The oscillators interact with each other locally, and control the periodic movement of each leg. This model can be regarded as a hierarchical structure consisting of two layers, the upper layer of brain and the lower layer of CPG. Brain layer performs global motion planning of the body, and CPG layer realizes the coordinated control of each legs.

By now, there are many studies on the locomotion of the multi-legged robot using decentralized control scheme, e.g. gait pattern generation [2, 3], an adaptive control [4, 5]. More recently, we have proposed such an autonomous decentralized control structure as CPG for a multi-legged robot to realize various locomotion patterns, considering robot's body conditions such as energy consumption. However all of these studies were restricted only to the straight-line walking. To adapt to the real world constraints, we have to realize not only the straight-line motion of robots but also the rotation and turning along a curve in the unknown environment.

Inspired by the study on animal's motion control structure, in this paper we propose a novel hierarchical control approach for the multi-legged robot. Here, the lower layer, corresponds to CPG, consists of subcontrollers which control its own leg according to a given vector field as well as the local interaction between the subsystems. The vector field is constructed by the weighted sum of base fields such as translation fields and rotation fields. The weights of each base fields are determined by the upper layer based on the received local environmental information. Such hierarchical control has two merits: First, the decentralized structure at the lower layer can considerably reduce the computation cost for each controller, com-

pared with the centralized case with using only one controller. Second, the upper layer can reflect the environmental information to the body movement faster than using only decentralized control. With only the decentralized controllers, each subsystem should plan the leg movement for the whole body movement to adapt to the environment.

2 Hierarchical Control Structure

In hierarchical control structure, it is important to study the load sharing problem between two layers by considering the total computation time as well as communication time. In the multi-legged robot's case, if the upper layer has to determine motion plans for every leg, then the computation cost will be too large. Conversely, if the upper layer just measures and sends the environmental information directly to the lower layer, then each sub controller at the lower layer has to spend long time for processing the common environmental information for all subsystems and also has to control each leg motion. Therefore, there exists a trade-off between centralized control system and decentralized one. Our hierarchical control structure for the multi-legged robot is shown in fig.1.

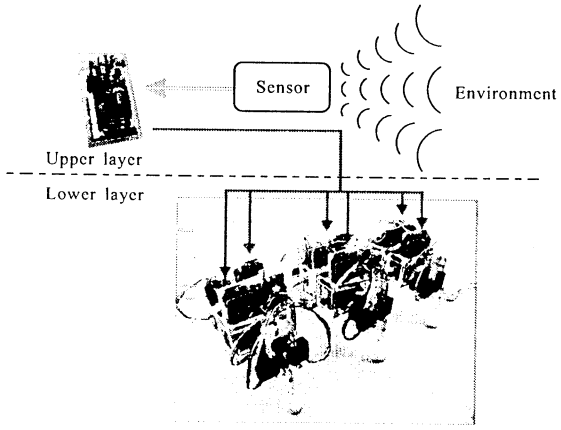


Fig.1: Hierarchical control structure

2.1 Upper Layer

The upper layer consists of the sensors, one processor and the communication channels. With receiving the sensor information about the environment, the processor plans the movement for the robot body. The results (as will be mentioned in next section) are then sent to the lower layer for each sub-controller to determine their own leg motion.

2.2 Lower Layer

The lower layer consists of six homogeneous subsystems. Each subsystem has one leg, one processor and three touch sensors detecting its neighbors, and it can interact with neighor subsystems.

The processor in each subsystem is given the information about a line in which it is arranged (left or right). And it can recognize the existence of the neighbor subsystems by touch sensors. Therefore it can know its own location within the robot as well as the direction of the robot's head. In addition, the processor of the subsystem receives two types of information. One is from the upper layer. Based on this information, each lower processor determines the swing direction and stride of its leg. The other information comes from the neighbor processors, and the interaction used this information keeps a gait pattern constantly. Each lower processor also has a periodic oscillator to determine the state, swing phase or stance phase of the leg. The phase of the oscillator is sent to its neighbor subsystems. Through local interaction, each processor adjusts its phase according to phase differences between neighbors based on the following gradient dynamics.

$$\frac{d\theta_i}{dt} = \omega - \sum_p \frac{\partial W(\psi_{ip})}{\partial \psi_{ip}} \qquad (1)$$

where θ_i is the phase of oscillator in ith subsystem, ω is constant angular velocity, $W(\psi_{ip})$ is a potential function as

$$W(\psi_{ip}) = -h\exp[-T\{1-\cos(\psi_{ip}-\psi_{ipd})\}] \qquad (2)$$

where T and h are constant values.

$$T = \frac{1}{2\sigma^2} \quad , \quad h = \frac{1}{\sqrt{2\pi\sigma^2}}$$

$\sigma = \frac{\pi}{2}$ corresponds to standard deviation. ψ_{ip} in equation (2) means the phase difference between ith subsystem and its neighbor. Subscript p means a direction where neighbors exist. And ψ_{ipd} is the target value of ψ_{ip}. In the gradient system with potential as equation (2), ψ_{ip} will converge to ψ_{ipd} [7]. Therefore, the phase of each oscillator and state of each leg is determined relatively against each other. After convergence, the second term on right side of equation(1) will be 0, and θ_i moves with constant angular velocity. Currently, ψ_{ipd}s for all i are set to π, it guarantee a tripod gait because each leg is in anti-phase against its neighbors.

3 Stride Control based on Vector Field

After determined the motion of a phase for each leg in above subsection, in this section we will use vector field to control the motion direction as well as stride for each leg such that the robot can perform translation and rotation.

3.1 Basic motion patterns of robot body

Any courses of the robot body can be decomposed into three types as shown in fig.2, fig.3 and fig.4, respectively. They are: translation, that is to go straight toward one direction, and rotations around a specific point. Fig.2 shows the translation case. In this figure, a rectangle and six dots are the robot body and coxae, respectively. And a double circle is the center of the robot body. In translation, every points on the robot move towards the same direction and distance. On the other hand, fig.3 and fig.4 shows the rotation

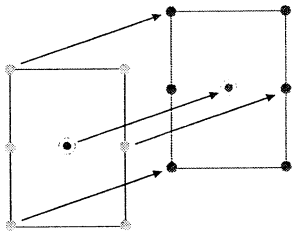


Fig.2: Translation of the robot

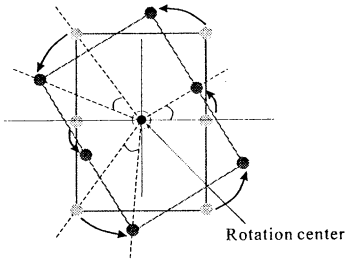


Fig.3: Rotation along the center of the robot's Body

cases. Fig.3 and fig.4 shows rapid turn and curve respectively. The difference between these two rotations is only the position of the rotation center. In these movements, any points on the robot rotate the same angle around the rotation center, but the distance is proportional to the distance from the center.

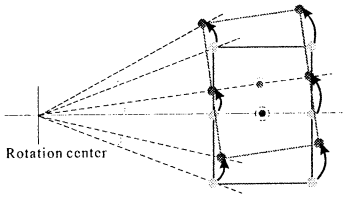


Fig.4: Rotation along a curve by the robot

All motion patterns can be realized by appropriate stride control of the legs. It is especially important to control the trajectory of the tips in stance phase. We introduce an angle between a line along the robot body and the trajectory of tip. Let this angle be γ (See fig.5). If γ of all legs are controlled correctly, the robot body will go along desired courses.

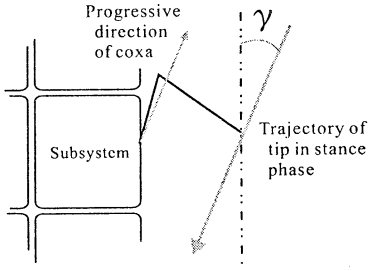


Fig.5: Angle of swinging for one leg in stance phase

3.2 Stride and direction of the leg motion

For three walking patterns of translation and rotations of the robot body, the stride and direction of each leg can simply be expressed by three base vectors.

We assume that all six processors at the lower layer already have these three base vector fields, and that both of stride and γ angle of each leg are determined by a weighted sum of these vector fields. A vector at one position of the vector field means the moving distance and direction of the corresponding point on the robot. When the robot goes straight, movement for any leg can be expressed by combining two base vector fields $\boldsymbol{A}$ and $\boldsymbol{B}$ as shown in fig.6. Two types of rotational movement (rapid turn and curve) is given by a base vector field $\boldsymbol{R}(X_c, Y_c)$. Here, (X_c, Y_c) is the position of rotation center. All complex movement can be obtained by weighted sum of three base vector fields as

$$\boldsymbol{V} = a_1\boldsymbol{A} + a_2\boldsymbol{B} + a_3\boldsymbol{R}(X_c, Y_c) \qquad (3)$$

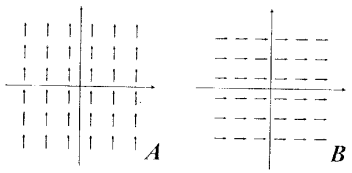


Fig.6: Vector fields for translation movement

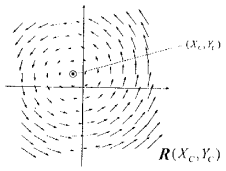


Fig.7: Vector field for rotation

Based on the environmental constraints, the upper processor mainly determines the weights (a_1,a_2 and a_3) as well as center parameters (X_c, Y_c) and sends them to all lower processors. Each lower processor calculates $\boldsymbol{V}$ and refers a vector at the corresponding leg's position. For example, in fig.8, six circles corresponds to six coxae of leg. Each lower processor refers a vector at each circle position in $\boldsymbol{V}$. The lower processors then use the vector to determine the stride and direction of the leg so that the robot body moves to the desired direction.

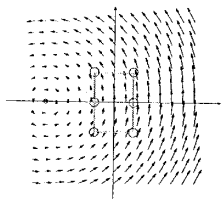


Fig.8: Vector field for each leg of the robot

4 Conclusions

A hierarchical structure is proposed for controlling a multi-legged robot to walk adaptively under unknown environmental constraints. Vector field is used to control the strides of each leg. The weights and parameters of each base field are determined at the upper layer based on the local environmental information. These parameters are then sent to all subsystems at lower layer to control the stride of legs. This approach has two merits: First, the calculation cost for each processor is much less than that for a single-processor-controlled robot. Second, the information about the environment is processed much faster comparing to the case when each processor plans its motion by only local interaction. Experimental results that a six-legged robot walks along a corner show the effectiveness of this approach.

References

[1] U.Bassler(1986),"On the definition of Central Pattern Generator and its Sensory Control" *Biological Cybernetics*,vol.54, pp.65-69.

[2] K.Inagaki, H.Kobayashi(1993), "Decentralized Autonomous Control for Quadruped Walking Robots and Their Synchronized Motion" *Journal of the Japan Society of Mechanical Engineers* Vol.59, No.564, pp2361-2367.

[3] H Yuasa, M Ito(1990)," Coordination of Many Oscillators and Generation of Locomotory" *Boilogical Cybernetics*, Vol.63, pp177-184.

[4] S.Ito, H.Yuasa, Z.W.Luo, M.Ito and D.Yanagihara(1998),"A mathematical model of adaptive behavior in quadruped locomotion" *Biological Cybernetics*,Vol.78,pp.337-347.

[5] S.Kimura, M.Yano and H.Shimizu(1993), "A self-organizing model of walking pattern of insects" *Biological Cybernetics*, Vol.69, pp183-193.

[6] T.Odashima, H.Yuasa, Z.W.Luo and M.Ito(1999), "Emergent generation of gait pattern for a myriapod robot system based on energy consumption(in Japanese)," *Journal of Robotics Society of Japan*,vol.17,No.8,pp.1149-1157.

[7] H.Yuasa, Y.Ito and M.Ito(1991),"Autonomous Distributed Systems whish Generate Various Patterns Using Bifurcation(in Japanese)" *Transactions of the Society of Instrument and Control Engineering*, vol.27,no.11,pp.1307-1314.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A Self-Organizing Visuo-Motor Map for a Redundant Manipulator with High Manipulability and Obstacle Free Poses

Nobuhiro Okada* Atsushi Yoshida** Eiji Kondo*
*Graduate School of Engineering,
Kyushu University
Fukuoka, 812-8581, JAPAN
**Kobe Shipyard & Machinery Works,
Mitsubishi Heavy Industries, LTD.
Kobe 652-8585 JAPAN

Abstract

A self-organizing visuo-motor map that maps from an image space into joint angle space is described. It determines the joint angles of a redundant manipulator so that the end effector of the manipulator reaches a target given in the image space. In this paper, we introduced estimate functions into the learning method of the visuo-motor map so that the manipulator achieves high manipulability and that it takes obstacle free poses in an environment with obstacles.

1 Introduction

Many researches have been done to control a manipulator resolving inverse kinematics problem using a neural network. Most of the networks are multilayered type so that they require supervised learning. On the other hand, self-organizing map suggested by Kohonen does not need it [1].

Zha et al. used a self-organizing map for a visuo-motor system [2]. They also applied it to a visuo-motor system in an environment that includes obstacles [3]. However, collisions between links of a manipulator and obstacles were not well considered. Furusho et al. used the manipulator's speed of approach to obstacles for obstacle avoidance [4]. Asada et al. proposed a method to construct a self-organizing visuo-motor map that controls a redundant manipulator [5]. The map achieved high manipulability around target positions. Maruki et al. combined the methods of Furusho et al. and Asada et al. to achieve high manipulability of a redundant manipulator in environments that includes obstacles [6]. We extended the network of Maruki et al. and introduced a potential field to avoid collisions between a manipulator and obstacles instead of the approach speed [7]. Using potential values of links of a manipulator in place of the value of an end effector, the network could prevent poses that might occur collisions between links and obstacles. However, we did not enough show the efficiency of the approach.

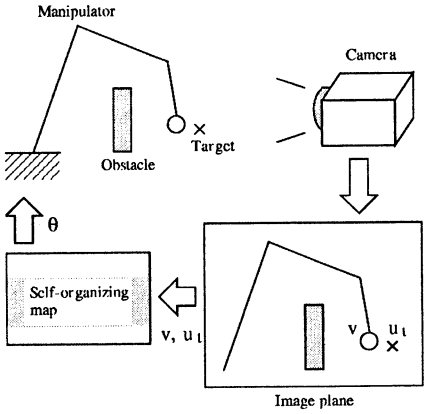


Figure 1: Outline of our visuo-motor system

In this paper, we will show how the visuo-motor map achieves high manipulability and obstacle free poses. Simulation and experimental results for 3 degrees of freedom redundant manipulator in a 2 dimensional space that includes a obstacle will be shown.

2 Our self-organizing map

Our visuo-motor system is illustrated in Fig.1. The system contains a redundant manipulator, a CCD camera, and a self-organizing map. The manipulator has 3 degrees of freedom and moves in a 2 dimensional space. The camera obtains the positions of the end effector $\boldsymbol{v}$ and the target $\boldsymbol{u_t}$ in the 2 dimensional space. The self-organizing map decides the joint angles of the manipulator $\boldsymbol{\theta}$ using input from the camera so that the end effector of the manipulator moves to the target.

The self-organizing map is composed of neurons that are distributed in the image space of the camera. It decides joint angles $\boldsymbol{\theta}$ as followings.

Each neuron N_i has following parameters.
$\boldsymbol{W}_i$: The position of the neuron.
$\boldsymbol{J}_i$: The Jacobi matrix from the joint space to the image space.
$\boldsymbol{\theta}_i$: The joint angle of the manipulator at $\boldsymbol{W}_i$.
$\boldsymbol{\xi}_i$: The gradient vector of the estimation function H.
When a target $\boldsymbol{u}_t$ is given in the image space, the neuron that has $\boldsymbol{W}_i$ nearest to the target is chosen. Then $\boldsymbol{\theta}^{out}$ is calculated (Eq.(1)).

$$\boldsymbol{\theta}^{out} = \boldsymbol{\theta}_i + \boldsymbol{J}_i^{+}(\boldsymbol{u}_t - \boldsymbol{W}_i) \tag{1}$$

Where $\boldsymbol{J}_i^{+}$ is a pseudo inverse matrix of $\boldsymbol{J}_i$. In the parameters of a neuron, $\boldsymbol{\xi}_i$ is used to make manipulator take poses that achieves high manipulability and that can avoid collisions.

As shown in Eq.(1), the map uses a linear projection as an approximation of the non-linear projection from the image space to the joint angle space. Therefore, time needed to calculate the $\boldsymbol{\theta}^{out}$ is suppressed, and an exact model of the visuo-motor system is not required. Moreover, the map can easily cope with a slight change of the visuo-motor system.

In actually, weighted sum of output of plural neurons around the target is used instead of Eq.(1).

$$\boldsymbol{\theta}^{out} = \frac{\sum_i g_n\{\boldsymbol{\theta}_i + \boldsymbol{J}_i^{+}(\boldsymbol{u}_t - \boldsymbol{W}_i)\}}{\sum_i g_n} \tag{2}$$

Where g_n is the weight and its value is determined according to the distance between the neuron and the target. It has a large value for a neuron that is near to the target, and has a small value for one that is far from the target.

3 Learning Algorithm of the Map

Before learning process, each neuron is given arbitrary initial parameters. Then the output joint angle $\boldsymbol{\theta}^{out}$ will lead the end effector to a wrong position and make the manipulator take a wrong pose. Following learning process amends the parameters.

In the learning process, an arbitrary target position in the image space is given to the visuo-motor system. The self-organizing map determines the joint angles $\boldsymbol{\theta}^{out}$ from the target position. The end effector of the manipulator does not, however, reach the target since the neurons in the map do not have appropriate parameters. Then, the map corrects the parameters for the end effector position in the image space obtained by the camera. The system repeats the above procedure for defined times.

Here, learning process on a real visuo-motor system will take a long time. Therefore it is useful to utilize the flexibility of the self-organizing map. Since self-organizing map can cope with changes between simulation and real system, a map that has learned by simulation will be applied to real visuo-motor system after less learning time.

For N-th iteration, the map amends the parameters as followings.
(1) A target position $\boldsymbol{u}_t^N$ is given in the image space. The position is transferred to the self-organizing map.
(2) The system sorts the neurons in the order of the distance between the target and $\boldsymbol{W}$ of each neuron.
(3) The map outputs the joint angles $\boldsymbol{\theta}_0^{out}$ obeying the Eq.2. Since the end effector does not reach the target position because of the errors of neuron parameters, the camera gets the position of the end effector $\boldsymbol{v}_0$.
(4) To reduce the error, change $\boldsymbol{\theta}^{out}$ obeying Eq.(3).

$$\boldsymbol{\theta}_1^{out} = \boldsymbol{\theta}_0^{out} + \frac{\sum_i g_n\{\boldsymbol{\theta}_i + \boldsymbol{J}_i^{+}(\boldsymbol{u}_t - \boldsymbol{v}_0)\}}{\sum_i g_n} \tag{3}$$

The camera also gets the position of end effector $\boldsymbol{v}_1$.
(5) The self-organizing map corrects the parameters using the methods mentioned in the following subsections.

3.1 Update of parameters in environments without obstacles

Each parameters of the neurons are updated using $\boldsymbol{u}_N$, $\boldsymbol{v}_0$, and $\boldsymbol{v}_1$.

$\boldsymbol{W}_i$ is updated by $\boldsymbol{W}_i^{N+1} = \boldsymbol{W}_i^N + \epsilon_W g_i(\boldsymbol{u}_t^N - \boldsymbol{W}_i^N)$. Where ϵ_W is the learning coefficient for $\boldsymbol{W}$, and it has a large value for early stages of learning process and has a small value for late stages. By updating $\boldsymbol{W}$ the neurons will be distributed in the image space.

$\boldsymbol{J}_i$ is updated by $\boldsymbol{J}_i^{N+1} = \boldsymbol{J}_i^N + \epsilon_J g_i \Delta\boldsymbol{J}_i^N$. Where ϵ_J is the learning coefficient for $\boldsymbol{J}$, and its value changes likely as ϵ_W. $\Delta\boldsymbol{J}_i^N$ is determined using Widrow–Hoff's learning rule. The estimation function is shown in Eq.(4).

$$E_j = \tfrac{1}{2}\|(\boldsymbol{v}_1 - \boldsymbol{v}_0) - \boldsymbol{J}_i^N(\boldsymbol{\theta}_1^{out} - \boldsymbol{\theta}_0^{out})\|^2 \tag{4}$$

Then $\Delta\boldsymbol{J}_i^N$ becomes as Eq.(5).

$$\Delta\boldsymbol{J}_i^N = \frac{(\boldsymbol{v}_{01} - \boldsymbol{J}_i^N\boldsymbol{\theta}_{01}^{out})\boldsymbol{\theta}_{01}^{outT}}{\|\boldsymbol{\theta}_{01}^{out}\|^2} \tag{5}$$

Where $\boldsymbol{v}_{01} = \boldsymbol{v}_1 - \boldsymbol{v}_0$, and $\boldsymbol{\theta}_{01}^{out} = \boldsymbol{\theta}_1^{out} - \boldsymbol{\theta}_0^{out}$. By updating $\boldsymbol{J}$ the map becomes to output more appropriate $\boldsymbol{\theta}^{out}$.

For environments without obstacles, neuron parameters $\boldsymbol{\theta}$ and $\boldsymbol{\xi}$ are updated as followings so that the manipulator gets high manipulability.

For the environments, $\boldsymbol{\xi}$ is the gradient vector of manipulability $(\partial H/\partial \boldsymbol{\theta})$. $\boldsymbol{\xi}$ is updated by $\boldsymbol{\xi}_i^{N+1} = \boldsymbol{\xi}_i^N + \epsilon_\xi g_i \Delta \boldsymbol{\xi}_i^N$. $\Delta \boldsymbol{\xi}_i^N$ is determined using Widrow–Hoff's learning rule likely as $\Delta \boldsymbol{J}_i^N$. The estimation function is shown in Eq.(6).

$$E_\xi = \tfrac{1}{2}\{(H_k - H_j) - \boldsymbol{\xi}_i^{NT}(\boldsymbol{\theta}_k^{out} - \boldsymbol{\theta}_j^{out})\}^2 \qquad (6)$$

Where j is the number of the neuron that has the 1st order for the distance between it and the target, and k is the number of the neuron that has the 2nd order. H_i is the manipulability for i–th neuron and is as Eq.(7).

$$H_i = \sqrt{\det(\boldsymbol{J}_i^N \boldsymbol{J}_i^{NT})} \qquad (7)$$

Therefore $\Delta \boldsymbol{\xi}_i^N$ becomes as following.

$$\Delta \boldsymbol{\xi}_i^N = \frac{(H_{jk} - \boldsymbol{\xi}_i^{NT}\boldsymbol{\theta}_{jk}^N)\boldsymbol{\theta}_{jk}^{NT}}{\|\boldsymbol{\theta}_{jk}^N\|^2} \qquad (8)$$

Where $H_{jk} = H_k - H_j$, and $\boldsymbol{\theta}_{jk}^N = \boldsymbol{\theta}_k^N - \boldsymbol{\theta}_j^N$.

$\boldsymbol{\theta}_i$ is updated by $\boldsymbol{\theta}_i^{N+1} = \boldsymbol{\theta}_i^N + \epsilon_\theta g_i \Delta \boldsymbol{\theta}_i^N$. Where ϵ_θ is the learning coefficient for $\boldsymbol{\theta}$, and its value changes likely as ϵ_W. In environments without obstacles, $\Delta \boldsymbol{\theta}_i^N$ becomes $\Delta \boldsymbol{\theta}_i^N = \boldsymbol{\theta}_i^{Desired} - \boldsymbol{\theta}_i^N$ where $\boldsymbol{\theta}_i^{Desired}$ is the desired joint angle. $\Delta \boldsymbol{\theta}_i^N$ is determined as followings so that the manipulator gets high manipulability.

The relation between speed of the joint angles $\boldsymbol{\theta}'$ and speed of position of the end effector position $\boldsymbol{x}'$ is $\boldsymbol{x}' = \boldsymbol{J}(\boldsymbol{\theta})\boldsymbol{\theta}'$. The general solution of $\boldsymbol{\theta}'$ becomes $\boldsymbol{\theta}' = \boldsymbol{J}^+\boldsymbol{x}' + (\boldsymbol{I} - \boldsymbol{J}^+\boldsymbol{J})\boldsymbol{k}$. Where $\boldsymbol{k}$ is an arbitrary n - vector. High manipulability of the manipulator is accomplished using the arbitrariness. Here, let $\boldsymbol{k}$ as $(\partial H(\boldsymbol{\theta})/\partial \boldsymbol{\theta})K$, or $\boldsymbol{\xi}K$. Where K is a positive coefficient that meets the condition where high manipulability is achieved and $\boldsymbol{\theta}_i'$ does not becomes too large. In our system, K is decreased by learning times. Then, $\boldsymbol{\theta}_i^{Desired}$ satisfies $\boldsymbol{\theta}_i^{Desired} - \boldsymbol{\theta}_o^{out} = \boldsymbol{J}_i^{N+}(\boldsymbol{W}_i^N - \boldsymbol{v}_o) + (\boldsymbol{I} - \boldsymbol{J}_i^{N+}\boldsymbol{J}_i^N)\boldsymbol{\xi}_i^N \boldsymbol{K}$. Therefore $\Delta \boldsymbol{\theta}_i^N$ becomes as following.

$$\Delta \boldsymbol{\theta}_i^N = \boldsymbol{\theta}_o^{out} - \boldsymbol{\theta}_i^N + \boldsymbol{J}_i^{N+}(\boldsymbol{W}_i^N - \boldsymbol{v}_o) + (\boldsymbol{I} - \boldsymbol{J}_i^{N+}\boldsymbol{J}_i^N)\boldsymbol{\xi}_i^N \boldsymbol{K} \qquad (9)$$

3.2 Update of parameters in environments with obstacles

For environments with obstacles, we introduce a potential field and replace H with Eq.(10).

$$H = \alpha_1 H_1 + \alpha_2 H_2 \qquad (10)$$

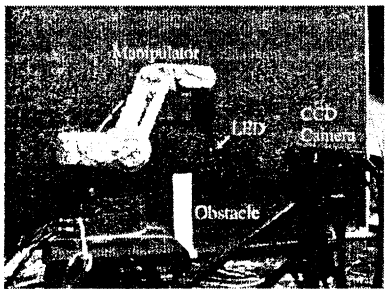


Figure 2: Our visuo-motor system

Where $H_1 = \sqrt{\det(\boldsymbol{J}_i^N \boldsymbol{J}_i^{NT})}$ and $H_2 = \sum_l (1.0 - \frac{D_0}{d_l + D_0})$. H_1 is the estimation function for manipulability, and H_2 is the function for environments with obstacles. Where d_l is the shortest distance from each link of the manipulator to obstacles, and D_0 is the prescribed value that defines the effective area of the potential.

Using Eq.(10), the map outputs $\boldsymbol{\theta}^{out}$ that achieves high manipulability and obstacle-free pose.

4 Simulation and experimental results

A photograph of our experimental system is shown in Fig.2. The manipulator is Mitsubishi Movemaster EX. The CCD camera sees the manipulator, and output from the camera is lead to a frame memory in the IBM–PC. The size of the frame memory is 512 × 512. The self-organizing visuo-motor map is composed in the PC. LEDs (light emission diode) are set on the the end effector and joints so that the system can find the positions of them in camera images. The CCD camera is also used to obtain shapes of obstacles. In the simulation and experiment, however, a simulated pattern was given as an obstacle shape.

A simulation result is shown in and Fig.3. In the simulations, the number of the neurons was 60, and 15,000 targets were given in the learning process. Figure 3 shows targets and poses of the manipulator after learning considering H. In the simulation, α_1 and α_2 of Eq.(10) were respectively 1.0×10^{-6}[(pixel/rad)^2] and 1.0[-]. In the figure, we can see that the manipulator moved its end effector to the targets, and that it took obstacle free poses. The transitions of the manipulability in the simulations are also shown in Fig.4. The graph shows that the manipulator achieved high manipulability by considering H.

An experimental result is shown in Fig.5. To shorten the time required for the learning process, the experiment used a self-organizing map of above simula-

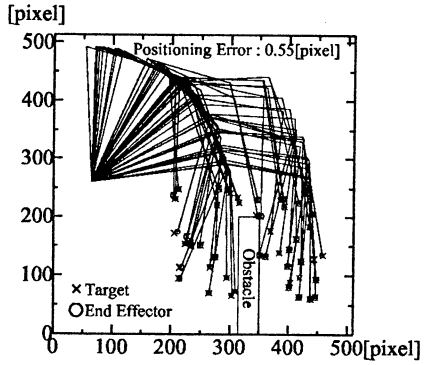


Figure 3: Poses with consideration of H

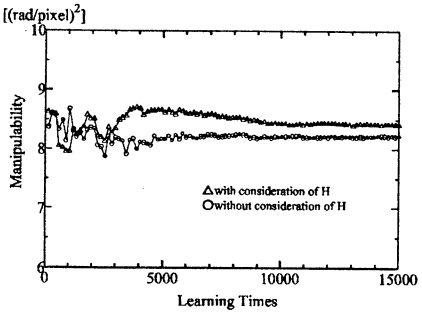


Figure 4: Transitions of the manipulability

tion as the initial state. Here, we used a fact that the self-organizing visuo-motor map can cope with slight changes of the visuo-motor system. Figure 5 shows targets and poses of the manipulator after learning. The number of targets given in the learning was 1,000 and the time required by the learning process was about 7[hours]. The average error between the end effector and targets was 3.5[mm]. The experimental result showed that the real visuo-motor system could reach the end effector to targets, achieve high manipulability, and take obstacle free poses. It also showed that using simulation before the experiment reduced the time required for the learning process.

5 Conclusion

In conclusion, we developed a self-organizing visuo-motor map. The map determines joint angles of a redundant manipulator so that the manipulator achieves high manipulability and takes an obstacle-free pose.

The obstacle used in this paper is one and has a simple shape. It is required to construct a self-organizing map that can deal with plural and complex shaped obstacles. Extension to a visuo-motor system in 3-dimensional space is also a future work.

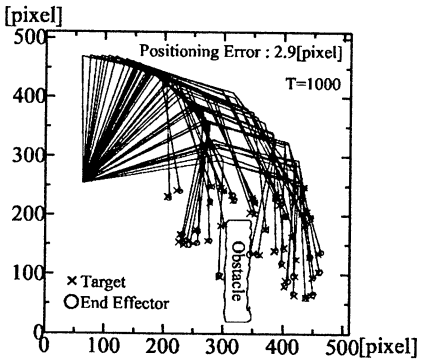


Figure 5: Experimental result

References

[1] Kohonen T (1988), Self-Organization and Associative Memory, Second Edition. Springer Series in Information Sciences, Vol.8

[2] Zha H, Onitsuka T, Nagata T (1995), Self-Organization Based Visuo-Motor Coordination for a Real Camera and Manipulator System. Proc. of the 1995 IEEE Int. Conf. on System, Man and Cybernetics (SMC'95), pp.3322–3327

[3] Zha H, Onitsuka T, Nagata T (1996), A Self-Organization Learning Algorithm for Visuo-Motor Coordination in Unstructured Environments. Proc. of Int. Symp. on Artificial Life and Robotics (AROB), pp.187–190

[4] Furusho J, Usui H, and Sano A (1988), Manipulability of Robotic Manipulators Considering the Influence of Obstacles (in Japanese). Journal of the Robotics Society of Japan, Vol.6, No.3, pp.12–20

[5] Asada K et al.(1995), Self-organization of a Task-oriented Visuo-Motor Map for a Redundant Manipulator. Technical Report of the Institute of Electronics, Information and Communication Engineers, Vol.NC95–76, pp.15–22

[6] Maruki Y et al. (1997), Self-Organization of Visuo-Motor Map Considering Obstacle (in Japanese). Technical Research Report of Common Center for Educational Research, Oita National College of Technology, 7th Issue, pp.8–12

[7] Okada N et al. (1999), A Self-Organizing Visuo-Motor Map for a Redundant Manipulator in Environments with Obstacles. Proc. of the 9th Int. Conf. on Advanced Robotics ('99 ICAR), pp. 517–522

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Simplified Geometric Models in Planning of Skill-Based Manipulation

°Akira NAKAMURA*
Kosei KITAGAKI*

*Electrotechnical Laboratory
1-1-4 Umezono, Tsukuba, Ibaraki, 305-8568 Japan

Tsukasa OGASAWARA**
Takashi SUEHIRO*

**Nara Institute of Science and Technology
8916-5 Takayamacho, Ikoma, Nara, 630-0101 Japan

E-mail: nakamura@etl.go.jp, ogasawar@is.aist-nara.ac.jp, kitagaki@etl.go.jp, suehiro@etl.go.jp

Abstract

Manipulator tasks such as assembly and disassembly can generally be divided into several motion primitives. We call these "skills" and explain how most manipulator tasks can be composed of skill sequences. Skills are also used to compensate for errors both in the geometric model and in manipulator motions. There are dispensable data in the shapes, positions and orientations of objects when achieving skill motions in a task. Therefore, we can derive simplified geometric models by considering both dispensable and indispensable data in a skill motion. This paper describes simplified geometric models in planning of skill-based manipulation.

Key words: manipulation skill, geometric model, planning

1. Introduction

In order for a robot to be useful in several fields, it is necessary to realize various tasks by using special techniques of manipulating robots. By analyzing human motions in tasks such as assembly and disassembly, movements were found to consist of several significant motion primitives. We called these "skills" and showed that most of the tasks of the manipulator can be composed of sequences of skills. That is, we demonstrated that robots can perform various human tasks by using the concept of skill.

Skills in which the contact states vary during assembly and disassembly tasks are particularly significant. We considered three skills of "move-to-touch," "rotate-to-level" and "rotate-to-insert," which play an important part in such tasks. First, the move-to-touch skill means the transition from "free" to "vertex-to-face" contact between a grasped object and another stationary object. The similar transition of keeping the contact in a different direction of motion is also included in this skill. Secondly, the rotate-to-level skill means the transition from "vertex-to-face" contact to "edge-to-face" contact. Thirdly, the rotate-to-insert skill means the motion of rotating the object obliquely into the hole in order to carry out the insertion. Most assembly tasks can be composed of these three skills.

It is possible to change simply the shapes of objects in planning of manipulation using skill technique. For example, let us consider unfastening task of a bolt using a wrench, composed of move-to-touch skills, rotate-to-insert skill and tilt motion. After attainment of move-to-touch skills, this task is always achieved by rotate-to-insert skill, independently of angle of a bolt. Therefore, it is not necessary to consider an orientation of bolt. When planning this task, we must carry out to derive the start region of move-to-touch skills. It can be derived by taking account of the inscribed and circumscribed circles of the hexagonal bolt. That is, a geometric model of the bolt can be regarded as concentric circles, so the shape is changed simply. Similarly, in a task of loosening a screw using a screwdriver, it is possible to change simply the shape of a screw in planning. There are many types of screws such as Phillips, Reed and Prince, Slotted, and Robertson, so we can show several simplified shapes of screws.

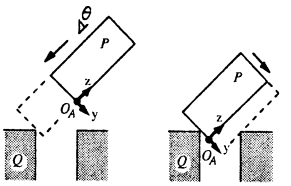
In this paper, we consider simplified geometric models in planning of skill-based manipulation and show the effectiveness in planning.

2. Manipulation Skills

2.1. Skill primitives

The motions of a manipulator in a task can be decomposed into several motion primitives, each of which has a particular target state. We call these primitives "skills," and in each primitive there is a transition from one state to another [1]-[4]. Most manipulation tasks can be achieved by a combination of skills.

Skills in which the contact states vary during assembly and disassembly tasks are particularly significant. In Reference [5], we considered three skills, "move-to-touch," "rotate-to-level" and "rotate-to-insert," which play an important part in such tasks. For simplicity, we assume that a grasped object,



(a) Move-to-touch$_f$ skill (b) Move-to-touch$_c$ skill

Fig. 1 Move-to-touch skills

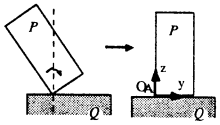


Fig. 2 Rotate-to-level skill

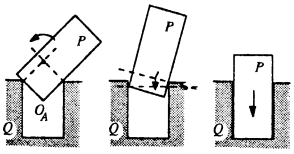


Fig. 3 Rotate-to-insert skill

another object and the hole are rectangular parallelepipeds.

(1) Move-to-touch Skill

The move-to-touch skill is defined as the transition of a grasped object P in a constant direction that continues until a contact with another object Q occurs (Fig. 1(a)). This skill is performed in velocity control mode. Not only the transition in free space (Fig. 1(a)), but also the sliding while keeping contact in a different direction of motion (Fig. 1(b)) is included in this skill. Figure 1 is drawn in a two-dimensional space. However, this skill is also defined similarly in the three-dimensional environment [6]. These two transitions are represented by the move-to-touch$_f$ skill and the move-to-touch$_c$ skill, respectively. The achievement of this skill can be detected by instantaneously increased resistance in the direction of the transition.

(2) Rotate-to-level Skill

The rotate-to-level skill is defined as rotation motion around either a contact point or a contact edge to match the face of the grasped object P with the face of another object Q. Figures 2 shows this skill in two-dimensional. This skill in three-dimensional space is shown in [6]. This skill is performed with a pushing force. The achievement of this skill can be detected by the change of the instantaneous center position.

(3) Rotate-to-insert Skill

We will explain this skill in two-dimensional space for simplicity. In an insertion task it is generally difficult to achieve the state in Fig. 3(b) directly when the clearance is small. The state in Fig. 3(a) is achieved first by using other skills, such as the skill sequence of Fig. 1(a) and (b). The state in Fig. 3(b) is then accomplished by gradually raising the object while maintaining contact as in Fig. 3(a). The rotate-to-insert skill is this motion of rotating the object P obliquely into the hole in another object Q to insert it accurately. 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. 3(c)). The rotate-to-insert skill in the three-dimensional space is defined as shown in [6]. The achievement of each rotate-to-insert skill is detected by the variation of the instantaneous center position.

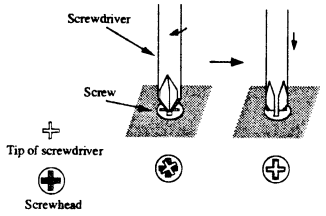


Fig. 4 Rotate-to-bite skill

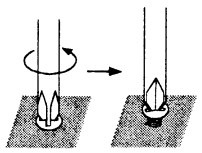


Fig. 5 Rotate-to-loosen skill

Furthermore, we will consider two skills related to the tasks in the examples mentioned later. In these two skills, we will explain using a cross-head screwdriver and screw according to tasks in examples.

(4) Rotate-to-bite Skill

This skill resembles the rotate-to-insert skill. The difference is the axis of rotation to achieve insertion. For simplicity, we will explain using an example of fitting a cross-head screwdriver to a screw as shown in Fig. 4. This skill is a rotation motion around the axis of the screwdriver to fit the tip of screwdriver to the flutes of the screw. This skill is performed with pushing force.

(5) Rotate-to-loosen Skill

This skill is defined as a rotation motion to loosen parts such as screws, bolts and nuts (Fig. 5). This is performed by matching the axes of rotation of a parts and a tool. If these axes do not correspond, the tool is moved to the position before execution of rotation. In this paper, we assume that this skill also includes the transition to remove an error before rotating.

2.2. Composition of Skill Sequence

A specific task is composed of sequences of skill primitives such as move-to-touch, rotate-to-level and rotate-to-insert skills. The skill sequences can be decided by several methods. We showed a method using variations of the number of contact points in skill primitives [5]. And, planning can be carried out by several techniques such as the backprojection method in which the trajectory is derived by taking into account the control errors of the manipulator grasping the object [5]. The initial position and orientation of the grasped object to achieve the task can be obtained.

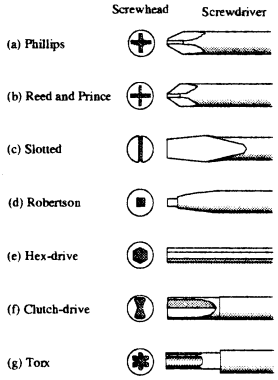


Fig. 6 Screwdrivers

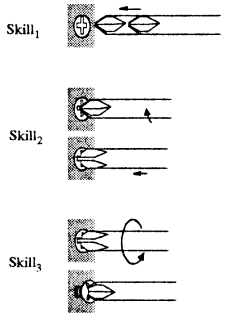


Fig. 7 Procedure of loosening with a cross-headed screwdriver

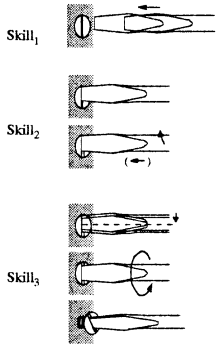


Fig. 8 Procedure of loosening with a slotted screwdriver

3. Simplified Geometric Models in Skill-Based Planning

In planning for manipulation, it is not necessary to use geometric models which express completely real objects. By constructing geometric models using only the data necessary for planning, it is possible to facilitate these processes. We call such robust models "simplified geometric models," which are defined as follows.

Simplified geometric models are geometric models composed of necessary and minimum data for shape, position and orientation to perform skill-based planning.

4. Examples

4.1. Task of loosening a screw using a screwdriver

We will consider the task of loosening a screw using a screwdriver. There are many kinds of screws and screwdrivers, and typical shapes are shown in Fig. 6. It is necessary to take into account the content of the task for each type. However, we will consider this task by unifying skill sequences in the following several examples.

Table 1 Simplified geometric models in the task of loosening a screw using a screwdriver

	Real Objects	Simplified Geometric Models		
		$Skill_1$	$Skill_2$	$Skill_3$
(a) Phillips			Not needed	Not needed
(b) Reed and Prince			Not needed	Not needed
(c) Slotted			Not needed	
(d) Robertson			Not needed	Not needed

We assume that the task of loosening a screw using a screwdriver is composed of the following skills as executed by a human.

Skill1 : Move-to-touch skill
Skill2 : Rotate-to-bite skill
Skill3 : Rotate-to-loosen skill

Next, we will consider simplified geometric models of screws used in modeling in *Skill1*, *Skill2* and *Skill3*. Simplified geometric models about each type of screw are shown in Table 1. We will explain it as follows.

4.1.1 Case of a cross-head screw

The procedure of this task for a cross-head screw is shown in Fig. 7. In general, most cross-head screws are the Phillips type (Fig. 6(a)). The Reed and Prince type (Fig. 6(b)) has little clearance.

Skill1 is achieved if the tip of the screwdriver gets into a groove in the neighborhood of the center of the screw-head. Simplified geometric model in *Skill1* is described by the shape of the groove (Table 1). The shape becomes a circle in order not to depend on the relative orientation between the screw and the screwdriver. The radius of the circle is derived from data of the shape of the screw. Therefore, the radius in a Reed and Prince type is smaller than that in a Phillips type. The region of the initial state of the move-to-touch skill of *Skill1* is obtained as set of trajectories projected inversely from this circle. Simplified geometric models in *Skill2*, *Skill3* are not needed since these can be executed continuously (Table 1).

4.1.2 Case of a slotted screw

The procedure of this task for a slotted screw is shown in Fig. 8. It is necessary to check the relative position and orientation between the screw and the screwdriver for achievement of *Skill1*. Simplified geometric model in *Skill1* is described by a flute (Table 1). There is a bit of room in the direction of the flute. For a slotted screw, *Skill2* is meaningless, so simplified geometric model in *Skill2* is not needed (Table 1). Since the axes of the rotations between the

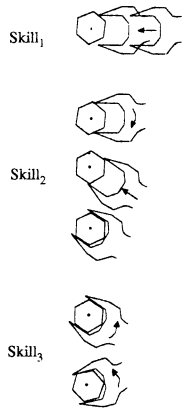


Fig. 9 Procedure of unfastening

screw and the screwdriver have to correspond for achievement of *Skill3*, data of the center of the screw is needed. Therefore, simplified geometric model in *Skill3* can be described by the center of the circle as shown in Table 1.

4.1.3 Cases of Robertson, hex-drive, clutch-drive and torx screws

The procedure of this task in these cases is the same as for a cross-head screw (Fig. 7). It is necessary to equalize the position and orientation between the screw and the screwdriver for achievement of *Skill1*. Simplified geometric models in *Skill1* become forms of grooves (Table 1). Simplified geometric models in *Skill2*, *Skill3* are not needed since these can be executed continuously (Table 1).

4.2. Task of unfastening a bolt using a wrench

Let us consider the task of unfastening a bolt using a wrench.

We assume that this task is composed of the following skills according to Reference [5] (Fig. 9).

Skill1 : Move-to-touch skill
Skill2 : Rotate-to-insert skill
Skill3 : Rotate-to-loosen skill

Next, we will explain simplified geometric models of the bolt used in modeling in *Skill1*, *Skill2* and *Skill3* (Table 2).

First, *M* and *m* is set as two distances in real object shown in Table 2. To derive the region of the initial state of the move-to-touch skill of *Skill1* as shown in [5], data of values *M*, *m* in shape and position of bolt are needed. However, it is not necessary to obtain the orientation of the bolt. Therefore, Simplified geometric model in *Skill1* is described by double circles (Table 2). Simplified geometric models in *Skill2*, *Skill3* are not needed since these can be executed continuously (Table 2).

Table 2 Simplified geometric models in the task of unfastening a bolt using a wrench

	Real Object	Simplified Geometric Models		
		$Skill_1$	$Skill_2$	$Skill_3$
Bolt			Not needed	Not needed

5. Conclusion

We have proposed simplified geometric models in planning of skill-based manipulation, and showed concretely examples of unfastening task of a screw and a bolt using tools. It is possible to carry out planning efficiently, since simplified geometric models are composed of the only indispensable data. In general, parts such as screws and bolts are small and the reliability of the model matching by part details such as matching by the grooves of screws and the hexagonal shape of bolts is low, so the orientation data for these parts are not always obtained correctly. Therefore, this technique using simplified geometric models is effective in the case that parts are too small to measure the data perfectly.

In the future, we will further study simplified geometric models for various skill-based tasks.

References

[1] Hasegawa T, Suehiro T, Ogasawara T, Matsui T, Kitagaki K, Takase K (1990), An integrated tele-robotics system with a geometric environment model and manipulation skills. In: Gotoh T, Kato I, Paul RP (eds) Proceedings of the IEEE International Workshop on Intelligent Robots and Systems (IROS '90), Tsuchiura, Japan, July 3-6, 1990, pp 335-341

[2] Suehiro T, Takase K (1990), Skill based manipulation system (in Japanese). J Robotics Soc Jpn 8:551-562

[3] 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

[4] Hasegawa T, Suehiro T, Takase K (1992), A model-based manipulation system with skill-based execution. IEEE Trans Robotics Autom 8(5):535-544

[5] Nakamura A, Ogasawara T, Suehiro T, Tsukune H (1996), Skill-based backprojection for fine motion planning. In: Asada M, Arai T, Kak A, Sandini G (eds) Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '96), Osaka, Japan, Nov. 4-8, 1996, pp 526-533

[6] Nakamura A, Ogasawara T, Suehiro T, Tsukune H (1998), Fine motion strategy in three-dimensional space using skill-based backprojection. Artificial Life and Robotics 2(3):134-137

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Study of Universal Learning Network with Branch Control

Qingyu Xiong, Kotaro Hirasawa
Intelligent Control Laboratory, Graduate School of Information Science & Electrical Engineering, Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan

Jinglu Hu, Junichi Murata
Intelligent Control Laboratory, Graduate School of Information Science & Electrical Engineering, Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan

Abstract

In this paper, Universal Learning Network with Branch Control (ULN with BC) is studied, which consists of basic networks and branch control networks. The branch control network can be used to determine which branches of the basic network should be used with a coefficient of relative strength. This determination depends on the inputs of the basic network. Therefore, by using the ULN with BC, locally functions distributed networks can be realized depending on the values of the inputs of the network. The proposed network is applied to some function approximation problems. The simulation results show that the ULN with BC exhibits better performance than the conventional networks with comparable complexity.

1 Introduction

In human's brain, it has been recognized that there are many special intellectual parts related to mathematics knowledge, music knowledge, gymnastic knowledge and so on, and each intellectual part is activated depending on the action of each knowledge. This is called "functions are locally distributed in the brain".[1]

But commonly used neural networks, especially layered neural networks, have the structure such that all input nodes and output nodes are connected to the intermediate nodes. Therefore, functions distribution of networks can not be realized in the conventional networks. Here, functions distribution means that only parts of the networks are activated depending on the input values of the network.

To obtain such a brain-like model in artificial neural networks for solving large-scale real-world problems, one of the most important things is how to divide a problem into smaller and simpler subproblems; and how to guide the training algorithm to realize the function distribution.

In this paper, an artificial functions distributed network is proposed and it is studied how much the proposed network can improve the performance of the networks compared to the conventional networks. The proposed network named Universal Learning Network with Branch Control (ULN with BC) is composed of two kinds of networks such as basic networks and branch control networks. And functions distribution of the basic network is materialized by connecting the branches of the basic networks with a coefficient of relative strength or disconnecting them following the instructions from the branch control networks in ULN with BC.

Modular networks [2][3] have similar module structures to ULN with BC. But, the structure of the modular networks is a little bit different from the one of ULN with BC. While the modules in modular networks are completely separated, ULN with BC has the module structure where the different modules have the identical nodes and branches, in other words, the modules in ULN with BC are overlapped. Therefore more compact networks are expected to obtain by ULN with BC.

2 Universal Learning Network with Branch Control

2.1 Basic Structure

Universal Learning Network with Branch Control (ULN with BC) is composed of basic networks and branch control networks (see Fig.1). And the relationship between the output of the branch control network and the hidden branch of the basic network is one to one correspondence. As a result, the output of the branch control network can control the connection of the corresponding hidden branches of the basic network with a coefficient of relative strength or can control the disconnection of the branches.

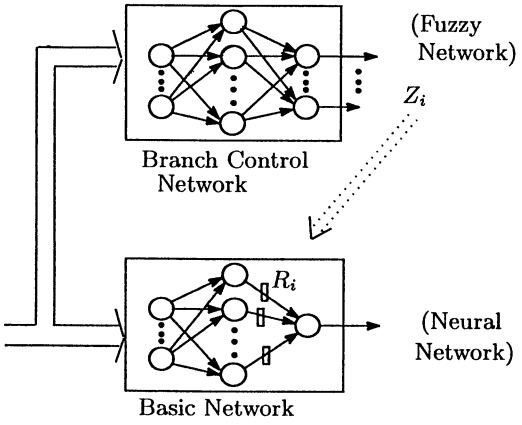


Figure 1: Structure of Universal Learning Network with Branch Control

In this paper, simulations were carried out adopting the following neural networks and fuzzy inference networks as the basic network and branch control network respectively.

- Neural Networks(x: input, y: output)

$$h_i = f(\omega_i^T x + b_i) R_i(Z_i) \quad (1)$$

$$y = f(\sum_{i=1}^{M} \lambda_i h_i + c) \quad (2)$$

$\omega_i, b_i, \lambda_i, c$: parameters of sigmoidal function;
f : sigmoidal function;
M : set of suffixes of intermediate nodes in neural networks;
R_j : a coefficient of relative strength determined by Z_j, $R_j \in [0,1]$;

- Fuzzy Inference Networks(x: input, Z_i: outputs)

$$Z_i = \frac{\sum_{q \in Q}[f_{xq}(x)\sigma_{iq}\mu_{iq}]}{\sum_{q \in Q}[f_{xq}(x)\sigma_{iq}]} \quad (3)$$

$$f_{xq}(x) = exp[-\frac{(x-\mu_{xq})^T \sum_{xq}^{-1}(x-\mu_{xq})}{2}] \quad (4)$$

$$f_{iq}(Z_i) = exp[-\frac{(Z_i-\mu_{iq})^2}{2\sigma_{iq}^2}] \quad (5)$$

$f_{xq}(x)$: membership function of the IF part of the fuzzy rule
$f_{iq}(Z_i)$: membership function of the $THEN$ part of the fuzzy rule
μ_{xq}: vector representing the mean of the IF part of the fuzzy rule
$\sum_{xq}^{-1}$: matrix representing the variance of the IF part of the fuzzy rule
μ_{iq}: parameter representing the mean of the $THEN$ part of the fuzzy rule, $\mu_{iq} \in [0,1]$
σ_{iq}^2: parameter representing the variance of the $THEN$ part of the fuzzy rule
Q: set of suffixes of fuzzy rules

Here, we set a threshold value Z_i^o for the output Z_i to determine which hidden branches should be connected or disconnected to the output layer in the basic network according to their one to one correspondence. And the relative strength R_i expressing how much the output Z_i exceeds the threshold value Z_i^o is determined by the following equation:

$$R_i(Z_i) = \begin{cases} \frac{Z_i - Z_i^o}{1 - Z_i^o} & \text{for } Z_i \geq Z_i^o \\ 0 & \text{for } Z_i < Z_i^o \end{cases} \quad (6)$$

This is because the contribution to the connectivity of the branches should be different according to how much Z_i exceeds Z_i^o (see Fig.2).

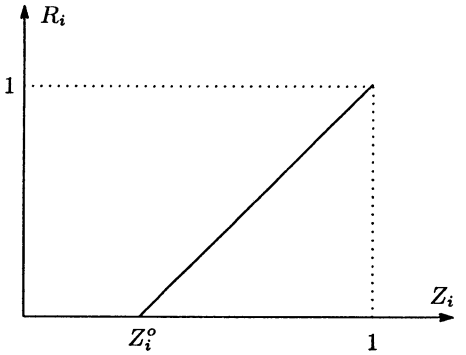


Figure 2: Relative strength R_i expressing how much Z_i exceeds Z_i^o

One of the reasons of using the fuzzy inference network as the branch control network is that neighboring points in the input space should be mapped also to the neighboring points in the output space in the branch control network in order to execute the branch control effectively.

Therefore, by adopting the fuzzy inference network as the branch control network with a coefficient of relative strength, a functions locally distributed network can be realized depending on the input values of the basic network.

2.2 Learning

As for learning, parameters of the basic network may be trained in the same way as the commonly used neural networks with supervised learning.

But, as branch connection with a coefficient of relative strength or disconnection of the basic network in ULN with BC depends on the input values of the network, it is difficult to train the parameters by using the gradient method like back propagation algorithm.

So, a kind of random search method named RasID [4] was used to train the parameters of the basic network. RasID is an abbreviation of Random Search with Intensification and Diversification, and executes the search for optimal parameters in a unified manner, where intensified search and diversified search are carried out systematically and effectively using the information on success and failure of the past search.

The following branch connectivity of the basic network can be adjusted by setting the threshold Z_i^o at an appropriate value:

$$\rho = \frac{m}{M} \tag{7}$$

where m is the numbers of the connected branches with a coefficient of relative strength, and M is the total numbers of hidden branches of the basic network. The branch connectivity ρ will be increased when Z_i^o takes a small value, and vice versa.

When functions locally distributed networks are used for many applications, it is generally expected that there exists a connectivity where the criterion function is minimized like Fig.3. This means that higher performance can be obtained than the commonly used networks by using the network with some branches being cut depending on the input values of the network. This fact has been already verified by using Learning Petri Network [5]-[8] developed in our laboratory.

The reason why the optimal connectivity exists in the networks is that it is appropriate to use the compact partial networks rather than the total big networks for the partial problems. In other words, it can be stated that since the compact partial networks have a small number of parameters, parameter training of the ULN with BC is expected to carry out smoothly.

3 Simulations of Universal learning Network with Branch Control

Simulations were carried out by adjusting the threshold Z_i^o in order to study the fundamental characteristics of connectivity ρ of ULN with BC.

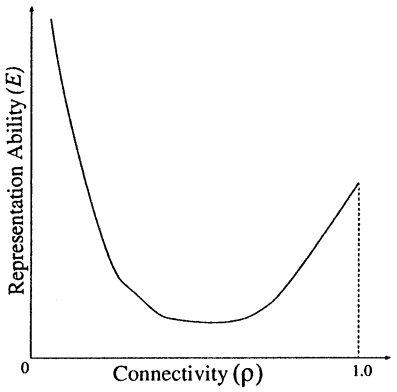


Figure 3: Relation between connection rate and generalization ability

3.1 Simulation Conditions

The parameters of the membership functions of the *IF* parts of the fuzzy rules $f_{xq}(x)$ are fixed in such a way that the input space of the network is appropriately covered with 5 fuzzy rules as follows:

$$\mu_{xq} = (0.25, 0.25), (0.25, 0.75), (0.75, 0.25), (0.75, 0.75), (0.5, 0.5)$$

$\sum_{xq}^{-1}$ is a diagonal matrix with its element being $\frac{1}{0.03}$, σ_{iq} is equal to 1.0 and μ_{iq} is random value in [0, 1].

And, simulations were done to study whether ULN with BC can realize the functions locally distributed network or not, and how much the representation and generalization ability can be improved by the functions distribution.

Simulation is executed by using the following two dimensional function approximation problem which are defined on $0 \leq x \leq 1, 0 \leq y \leq 1$ (see Fig.4). This function has an uniform characteristic on all areas under consideration.

$$f(x,y) = 0.475((1.35 + exp(x)sin(13.0(x-0.6)^2)\cdot exp(-y)sin(7y)) - 2.5) - 0.2$$

The basic network is a three layered neural network with the number of the intermediate nodes being 20.

Another parameters in the basic network are initialized with random numbers.

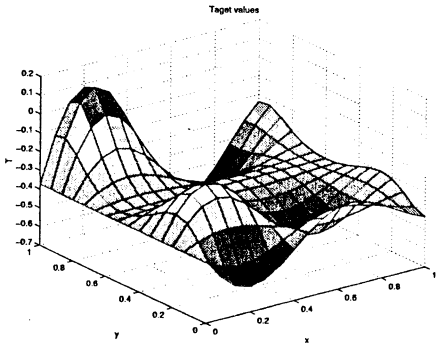


Figure 4: Target function to be approximated

3.2 Simulation Results

Tab.1 and Fig.5 show how the performance of the ULN with BC changes as the connectivity ρ varies by adjusting the threshold Z_i^o. (They are the average results over 10 times for each threshold value) It is clearly stated that it has the tendency to have optimal connectivity, around which the best performance is obtained like Fig.3.

Table 1: Sum Squared Error for Different Thresholds

Z_i^o	0.0	0.1	0.2
ρ	1.0	0.95	0.88
SSE	0.1222	0.0784	0.0773
Z_i^o	0.3	0.4	0.5
ρ	0.78	0.61	0.48
SSE	0.1255	0.1327	0.1694

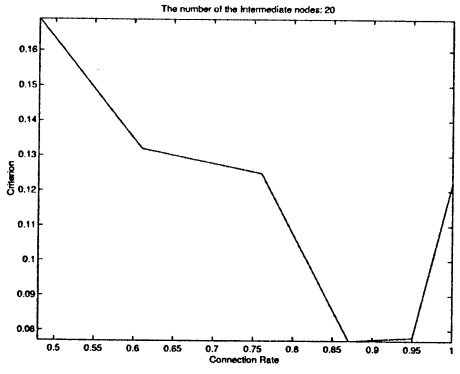


Figure 5: SSE for different thresholds

4 Conclusions

In this paper, a new type of functions locally distributed network is proposed, where the branch control network can control the branch connection and disconnection of the basic network with a coefficient of relative strength. And from simulations, it has been clarified that there exists an optimal connectivity where the best performance of the network can be obtained, which means that commonly used neural networks do not always exhibit the best performance. This fact is not well known in the neural network community.

References

[1] T. Sawaguchi: *Brain Structure and Evolution*, Kaimeisha, 1989 (in Japanese)

[2] R. A. Jacobs and M. I. Jordan, "Learning piecewise control strategies in a modular neural network architecture", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 23, pp.337-345, 1993.

[3] M. I. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and the EM algorithm", *Neural Computation*, Vol. 6, pp.181-214, 1994.

[4] J. Hu, K. Hirasawa and J. Murata, "RasID - Random Search for Neural Network Training", *Journal of Advanced Computational Intelligence*, Vol. 2, No. 4, pp.134-141, 1998.

[5] K. Hirasawa, M. Ohbayashi, S. Sakai and J. Hu, "Learning Petri Network and Its Application to Nonlinear System Control", *IEEE Transactions on Systems, Man, and Cybernetics Part B*, Vol. 28, No. 6, pp.781-789, 1998.

[6] K. Hirasawa, S. Oka, M. Ohbayashi, S. Sakai and J. Murata, "Locally Functions Distributed Network -Learning Petri Network- based on Petri Network", *Transactions of the Society of Instrument and Control Engineers of Japan*, Vol. 32, No. 6, pp.241-250, 1996.(in Japanese)

[7] M. Ohbayashi, K. Hirasawa, S. Sakai and J. Hu, "An Application of Learning Petri Network to Nonlinear System Control", *Transactions of the Institute of Electrical Engineers of Japan*, Vol. 115-C, No. 6, pp.875-881, 1998.(in Japanese)

[8] K. Hirasawa, M. Ohbayashi, J. Murata, C. Jin, S. Oka, S. Sakai, Y.Shitamura, "Modeling of Brain's Functions Distribution by Petri Network", *Transactions of the Institute of Electrical Engineers of Japan*, Vol. 115-C, No. 5, pp.719-727, 1995. (in Japanese)

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Neural-Network Controller of Articulated Robot Arm with Interference for High-precision Contour Control

○ T.ZHANG, M.NAKAMURA, S.GOTO
Dept. of Advanced Systems Control
Saga University
Honjomachi, Saga 840-8502
E-mail:zhangtao@cntl.ee.saga-u.ac.jp

N.KYURA
Dept. of Electrical Engineering
Kinki University (in Kyushu)
11-6 Kayanomori, Iizuka 820-8555

Abstract

In this paper, a new contour controller by using Gaussian neural network (GNN) was proposed to overcome the interference between robot links. Firstly, the principle of constructing the GNN controller and approximating the interference with the Euler-Lagrange model of articulated robot arm was presented. Then the actual input/output data of articulated robot arm was used for training the GNN to accurately represent the inverse dynamics of articulated robot arm with interference. Finally, through the simulation and experiment, it was verified that the precision of the contour control has been improved by the proposed method comparing with the conventional method.

Key words Contour Control, Articulated Robot Arm, Interference, Gaussian Neural Network

1 INTRODUCTION

At present, there are many applications of artificial neural network in the field of robotics because of its many attractive intelligent features, such as ability of learning, approximation of function, robustness, adaptiveness, and so on, which are capable of solving the problems with uncertain factors. The contour control is one of the important working patterns for articulated robot arms which have been widely used in industry. Facing to this kind of control, high efficiency with high precision is a universal expected performance in operation of articulated robot arm. However, with the increasing of speed, the accuracy of the contour control performance becomes low, which causes the actual trajectories much distortion. One of the important reasons is the interference between robot links. The so-called interference refers to the mutually coupling between robot links. The main reasons of interference come partly from the inertial moment and partly from the other forces, such as Coridis and centrifugal force. From the experience of view, interference between robot links can not be neglected when the precision criterion is bigger than a certain threshold or the speed is required to be much higher. Therefore, in this paper, a new contour control method by using Gaussian neural network was proposed to solve this problem.

2 METHOD

2.1 System architecture of the contour control system

The system architecture of the contour control of articulated robot arm for each link is depicted in Fig.1(a) which contains the GNN controller, model of articulated robot arm and kinematics. The model of articulated robot arm represents the actual robot arm with servo system. Kinematics can be described as

$$\begin{cases} x = l_1 \sin q_1 + l_2 \sin (q_1 + q_2) \\ y = l_1 \cos q_1 + l_2 \cos (q_1 + q_2) \end{cases} \quad (1)$$

where l_1, l_2 are the length of robot links, q_1, q_2 are the position of the end-effectors in joint co-ordinates, and x, y are the position of the end-effectors in Cartesian co-ordinates. Due to using feedforward control, it is not necessary to feedback the output of the system to the GNN controller and it can achieve the real-time control in the high-speed motion.

To achieve high accuracy of the contour control for each link, the nonlinear separation strategy was adopted proposed by Nakamura *et al*[1]. The controller of articulated robot arm was divided into two parts: nonlinear statics and linear dynamics (see Fig.1(b)). The so-called nonlinear statics refers to the inverse kine-

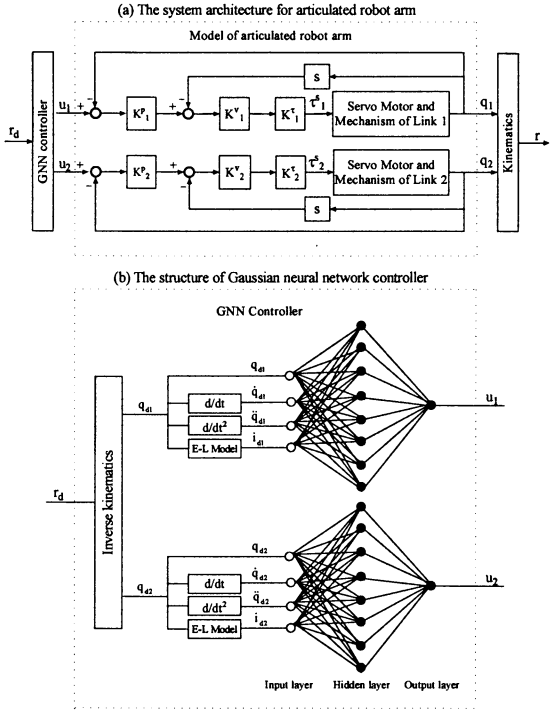


Figure 1: (a)The system architecture for articulated robot arm; (b)The structure of Gaussian neural network controller.

matics described as

$$\begin{cases} q_1 = \arctan\dfrac{x}{y} + \arccos\dfrac{(l_1)^2 - (l_2)^2 + x^2 + y^2}{2l_1\sqrt{c_2 + y_2}} \\ q_2 = \pi - \arccos\dfrac{(l_1)^2 + (l_2)^2 - x^2 - y^2}{2l_1 l_2} \end{cases} \quad (2)$$

Inverse kinematics solves for the arm configuration in joint co-ordinates, given the position and orientation of the end-effectors in Cartesian co-ordinates. The so-called linear dynamics refers to the proposed GNN which is designed for compensating the system delay dynamics and eliminating the influence of interference. Based on the nonlinear separation strategy, the control process for articulated robot arm is easily understandable and explicit.

2.2 Construction of Gaussian neural network controller

The Gaussian neural network is a kind of multilayer feedforward networks with Gaussian potential function (GPF)[2]. It is capable of approximating a many-to-one continuous function by a potential field synthesized over the domain of the input space by a number of GPF units. To construct the inverse dynamics of articulated robot arm with interference by using GNN, the Euler-Lagrange equations[3] were adopted to model the articulated robot arm with interference for each link described as

$$\begin{aligned} K_1^p K_1^v K_1^\tau u_1 &= \{m_1(l_1^G)^2 + m_2[(l_1)^2 + (l_2^G)^2] \\ &\quad + \frac{m_1(l_1)^2}{3} + \frac{m_2(l_2)^2}{3} + (k_1)^2 J_1^M\}\ddot{q}_1 \\ &\quad + K_1^v K_1^\tau \dot{q}_1 + K_1^p K_1^v K_1^\tau q_1 + i_1 \\ K_2^p K_2^v K_2^\tau u_2 &= [m_2(l_2^G)^2 + \frac{m_2(l_2)^2}{3} + (k_2)^2 J_2^M]\ddot{q}_2 \\ &\quad + K_2^v K_2^\tau \dot{q}_2 + K_2^p K_2^v K_2^\tau q_2 + i_2 \end{aligned} \quad (3)$$

And the interference can be described as

$$\begin{aligned} i_1 &= (2m_2 l_1 l_2^G \cos q_2)\ddot{q}_1 + [m_2(l_2^G)^2 + m_2 l_1 l_2^G \cos q_2 \\ &\quad + \frac{m_2(l_2)^2}{3}]\ddot{q}_2 - 2m_2 l_1 l_2^G \dot{q}_1 \dot{q}_2 \sin q_2 \\ &\quad - m_2 l_1 l_2^G \dot{q}_2^2 \sin q_2 \\ i_2 &= [m_2(l_2^G)^2 + m_2 l_1 l_2^G \cos q_2 + \frac{m_2(l_2)^2}{3}]\ddot{q}_1 \\ &\quad + m_2 l_1 l_2^G \dot{q}_1^2 \sin q_2 \end{aligned} \quad (4)$$

where $K_j^p (j = 1, 2)$ is the position loop gain, $K_j^v (j = 1, 2)$ is the velocity loop gain, $K_j^\tau (j = 1, 2)$ is the torque coefficient, $m_j (j = 1, 2)$ is the mass of robot link, $k_j (j = 1, 2)$ is the geer ratio, $J_j^M (j = 1, 2)$ is the inertial coefficient, $l_j (j = 1, 2)$ is the length of robot link, and $l_j^G (j = 1, 2)$ is the length between the axis and the centre of gravity.

When we replace the $[q_j(t)\ \dot{q}_j(t)\ \ddot{q}_j(t)](j = 1, 2)$ as $[q_j^d(t)\ \dot{q}_j^d(t)\ \ddot{q}_j^d(t)]$ in equation (3) and (4), where $q_j^d(t)$, $\dot{q}_j^d(t)$ and $\ddot{q}_j^d(t)$ denotes the objective trajectory, velocity and acceleration in the joint co-ordinate respectively, then the inverse dynamics of articulated robot arm with interference can be obtained.

Although it is capable of getting the inverse dynamics of articulated robot arm with interference by above equations, the GNN will be adopted to express the inverse dynamics of articulated robot arm with interference through the learning from the actual experimental data of articulated robot arm instead of them. There are three reasons: 1) The ideal values of servo system parameters, such as the position loop gain K^p, the velocity loop gain K^v and the torque coefficient K^τ, could not be obtained because of the variance of the manufacturing products; 2) The limitation of position, velocity, acceleration and interference, which is very significant in industry, are not considered; 3) The interference expressed by the Euler-Lagrange model exists error between the model and actual values.

According to the above mathematical expression of inverse dynamics of articulated robot arm with interference, the structure and initial parameters of GNN will be determined. The input of the network is $\mathbf{x} = [q(t)\ \dot{q}(t)\ \ddot{q}(t)\ i(t)]^T$, where $q(t)$, $\dot{q}(t)$, $\ddot{q}(t)$, $i(t)$

represent the objective trajectory, velocity, acceleration and approximated interference respectively, and the output of the network is $u(t)$. To determine the initial parameters of GNN, one-input, one-output and two-Gaussian-unit sub-networks are selected for each element of input vector. Therefore, there are four such kinds of sub-networks whose input denote different elements of $\mathbf{x}$ and whose output denote part of output of $u(t)$. Then they are combined for expressing the inverse dynamics of articulated robot arm with interference. If define z to represent each element of $\mathbf{x}$, then in each sub-network, the output can be expressed as

$$\phi(\mathrm{z}) = \mathrm{w}_1 \exp[-\frac{(\mathrm{z}-\mu_1)^2}{2(\sigma_1)^2}] + \mathrm{w}_2 \exp[-\frac{(\mathrm{z}-\mu_2)^2}{2(\sigma_2)^2}] \quad (5)$$

where w_j, μ_j and $\sigma_j (j = 1, 2)$ denotes the weight, mean, marginal standard deviation of the jth Gaussian unit of the sub-network, respectively.

In order to express the each item in the equations of inverse dynamics and add into the limitation for each input element, the output of each sub-network should be satisfied the condition as $\phi(z) \approx az$ inside of $|z| < Z_{max}$ and $\phi(z)$ converging to zero outside of $|z| < Z_{max}$, where Z_{max} denotes the relative maximum of z and a is a positive constant as the coefficient of input element in the equations of inverse dynamics. Hence, the mean μ_1 is chosen as Z_{max} and μ_2 is chosen as $-Z_{max}$ because the function $\phi(z)$ needs to convergence to zero outside of $|z| < Z_{max}$. The function $\phi(z)$ is approximated as

$$\begin{aligned}\phi(z) \approx\ & w_1 \exp[-\frac{(\mu_1)^2}{2(\sigma_1)^2}] + w_2 \exp[-\frac{(\mu_2)^2}{2(\sigma_2)^2}] \\ & + \{\frac{w_1\mu_1}{(\sigma_1)^2} \exp[-\frac{(\mu_1)^2}{2(\sigma_1)^2}] + \frac{w_2\mu_2}{(\sigma_2)^2} \exp[-\frac{(\mu_2)^2}{2(\sigma_2)^2}]\} z\end{aligned} \quad (6)$$

by the Taylor expansion of equation (5).

Besides, the standard deviation σ_i is selected as $0.57\mu_i$ from the point of view of the sub-network output closing to az and $w_1 = -w_2 = w$ for the unique input in each sub-network[4]. Therefore, comparing with az and based on the following conditions

$$\begin{cases} w_1 = -w_2 = w \\ \mu_1 = -\mu_2 = Z_{max} \\ \sigma_1 = 0.57\mu_1 \\ \sigma_2 = 0.57\mu_2 \end{cases} \quad (7)$$

the relationship between the weight of output layer of sub-network and the constant a is obtained as

$$\mathrm{w} = \frac{(0.57)^2 \mathrm{a} \mathrm{Z}_{\max}}{2} \exp[\frac{1}{2 \times (0.57)^2}] \quad (8)$$

Besides, for the input of considered sub-network, the mean μ, standard deviation σ and correlation coefficient h for the Gaussian units in other sub-networks are defined as zero, a large positive constant and zero respectively because there are no any relationship among each sub-network at the initial state.

In order to learn the true characteristics of articulated robot arm with interference, the actual sampled input/output data generated from articulated robot arm are adopted as the teaching patterns. The back-propagation method is used for the learning algorithm. The parameters of GNN are updated for each sampled teaching pattern and the algorithm stops when the parameters converge.

3 RESULTS

In order to verify the effectiveness of the proposed method, both simulation and experiment have been done under the same conditions. For the simulation, the articulated robot arm was emulated by the Euler-Lagrange model of articulated robot arm with interference. For the experiment, a Performer MK3S industrial articulated robot arm produced by Yahata Co., Japan, was used.

Based on the performances and parameters of Performer MK3S, the following conditions were given: 1) the position loop gains: $K_1^p = K_2^p = 25$[1/s]; 2) the velocity loop gains: $K_1^v = K_2^v = 150$[1/s]; 3) the torque constants: $K_1^\tau = 0.104$[Nms^2/rad], $K_2^\tau = 0.061$[Nms^2/rad]; 4) the geer ratios: $k_1 = 160$, $k_2 = 161$; 5) the inertia coefficients: $J_1^M = 4.0 \times 10^{-7}$[Nms^2], $J_2^M = 2.7 \times 10^{-7}$[Nms^2]; 6) the mass of links: $m_1 = 2.86$[kg], $m_2 = 2.19$[kg]; 7) the length of links: $l_1 = 0.25$[m], $l_2 = 0.215$[m]; 8) the length between the axis and the centre of gravity: $l_1^G = 0.11$[m], $l_2^G = 0.105$[m]; 9) the sampling interval : 0.001[s]. Besides, according to the experience of actual operation and consideration of safety, the allowance region of robot arm in the joint co-ordinate was selected as 4.72[rad] and 6.28[rad] for the link 1 and link 2. The allowance velocity was selected as 0.375[rad/s] and 0.5[rad/s]. The allowance acceleration was selected as 0.54[rad/s^2] and 0.72[rad/s^2]. Since the maximal interference was smaller than 10% of the maximal acceleration through simulation and experiment, therefore, the region of interference was defined as 0.03[rad/s^2] and 0.02[rad/s^2] for link 1 and link 2 by approximating the interference of the training trajectory given in equation (9) with the Euler-Lagrange model.

The training trajectory was defined as (see Fig.2 (a))

$$\begin{aligned} r_{d_1} &= R\cos(0.2\pi t) + R\cos(0.2\pi t)/5 \quad (0 \le t \le 10) \\ r_{d_2} &= R\sin(0.2\pi t) + R\sin(0.2\pi t)/5 \quad (0 \le t \le 10) \end{aligned} \quad (9)$$

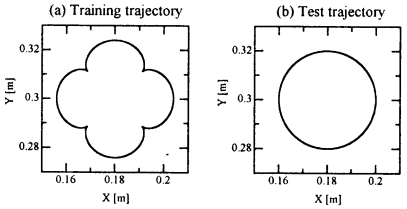


Figure 2: The training and test trajectories in the Cartesian co-ordinate

The radius R of the training trajectory was 2[cm]. After sampling, the training trajectory and its relative velocity and acceleration calculated by Euler equation, and the interference approximated by Euler-Lagrange model, were put into the GNN controller with initial parameters. Then the control signals from the GNN controller were put into the actual articulated robot arm in the experiment or the model of articulated robot arm in the simulation. Finally, the training patterns were generated and they were consisted of the input/output data of articulated robot arm. Besides, the training rate η was selected as a small value of 0.001 because the GNN with the initial parameters closed to the inverse dynamics of articulated robot arm.

In order to verify the GNN controller, a circle was defined as the test trajectory and described as (see Fig.2 (b))

$$\begin{aligned} r_{d_1} &= R\cos(0.2\pi t) \quad (0 \leq t \leq 10) \\ r_{d_2} &= R\sin(0.2\pi t) \quad (0 \leq t \leq 10) \end{aligned} \tag{10}$$

The radius of the objective circle was also 2[cm]. For comparison, the modified taught data (MTD) method proposed by Goto *et al.*[5] was also adopted to make simulation and experiment with the same conditions and test trajectory. Fig.3 shows the contour control simulation and experiment results of articulated robot arm with interference.

4 CONCLUSIONS

A contour controller for articulated robot arm with interference by using Gaussian neural network has been proposed. With this method, not only the inverse dynamics of the system has been compensated, but also the influence of interference between robot links has been eliminated. Through the simulation and experiment, the effectiveness of the proposed method was verified.

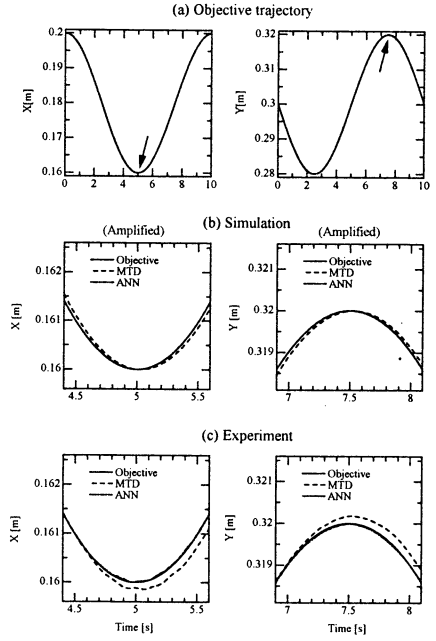


Figure 3: The verification by test trajectory

References

[1] M.Nakamura, S.Goto and T.Sugi, "A Methodology for Designing Controllers for Industry Systems based on Nonlinear Separation Model and Control", *Control Engineering Practice*, vol.7, pp.347-356, 1999.

[2] S.Lee and R.M.Kil, "A Gaussian Potential Function Network with Hierarchically Self-organizing Learning", *Neural Networks*, vol.4, pp.207-224, 1991.

[3] K.S.Fu, R.C.Gonzalez and C.S.G.Lee, "Robotics: Sensing, Vision and Intelligence", *New York: McGraw-Hill Book Co.*, 1987.

[4] S.Goto, M.Nakamura and N.Kyura, "Accurate Contour Control of Mechatronic Servo Systems Using Gaussian Networks", *IEEE Trans. On Industrial Electronics*, vol.43, pp.469-476, 1996.

[5] S.Goto, M.Nakamura and N.Kyura, "Modified taught data method for industrial mechatronic servo-controller to achieve accurate contour control performance", *IEEE/ASME Int. Conf., on Advanced Intelligent Mechatronics (AIM)*, 525B, June, 1997.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Chaos control by a stochastic analysis on recurrent neural networks

Masao Sakai
Graduate School of Eng.,
Tohoku University,
05, Aoba, Aramaki, Aoba-ku,
Sendai, 980-8579, JAPAN

Noriyasu Honma
College of Medical Sciences,
Tohoku University,
2-1, Seiryo-machi, Aoba-ku,
Sendai, 980-8575, JAPAN

Kenichi Abe
Graduate School of Eng.,
Tohoku University,
05, Aoba, Aramaki, Aoba-ku,
Sendai, 980-8579, JAPAN

(sakai,homma,abe@abe.ecei.tohoku.ac.jp)

Abstract

This paper demonstrates that the largest Lyapunov exponent λ of recurrent neural networks can be controlled by a gradient method. The method minimizes a square error $e_\lambda = (\lambda - \lambda^{obj})^2$ where λ^{obj} is desired exponent. The λ can be given as a function of the network parameters P such as connection weights and thresholds of neurons' activation. Then changes of parameters to minimize the error are given by calculating their gradients $\partial\lambda/\partial P$.

In a previous paper, we derived a control method of λ via a direct calculation of $\partial\lambda/\partial P$ with a gradient collection through time. This method however is computationally expensive for large-scale recurrent networks and the control is unstable for recurrent networks with chaotic dynamics.

Our new method proposed in this paper is based on a stochastic relation between the complexity λ and parameters P of the network configuration under a restriction. Then the new method allows us to approximate the gradient collection in a fashion without time evolution. This approximation requires only $O(N^2)$ run time while our previous method needs $O(N^5T)$ run time for networks with N neurons and T evolution. Simulation results show that the new method can realize a "stable" control for large-scale networks with chaotic dynamics.

1 Introduction

Recurrent neural networks, consisting of units connected with each other, have higher degree of parameter freedom compared with that of feedforward neural networks composed of the same number of units. Harnessing the dynamics of complicated interactions among the units, the recurrent network is expected to become a useful model for identifying and controlling the nonlinear complex dynamical systems[7].

Most of learning algorithms for the recurrent networks are based on the algorithms for the feedforward networks. For example, Jordan has proposed new type of recurrent networks which can be learned by the well-known back-propagation algorithm using the supervisory signals as the feedback signals[6]. In this case, the Jordan's recurrent networks can approximate the input-output function of the target systems even if the functions are nonlinear[5].

However there is no guarantee that dynamical complexity of the recurrent networks converges to the target complexity[2][8]. This means results with respect to the behavior of the actual target systems could be different from the estimated knowledge based on the recurrent networks learned by conventional methods. Especially the chaotic systems aren't captured by the conventional learning algorithm since when the neural model begins to learn the system in an accurate form, then it is itself chaotic and update of the weights are therefore unstable.

Principle and Kuo have proposed a complexity control method which updates the weights with a forgetting function given by the largest Lyapunov exponent for feedforward networks[9]. For recurrent networks Deco and Schürman have reported that the dynamical complexity can be learned by a stochastic "sample-by-sample" update of the weights with the forgetting function[1]. There is however no direct formulation for controlling the largest Lyapunov exponent of recurrent networks.

In a previous paper, we proposed a gradient-based control method of the largest Lyapunov exponent [2]. The method gave a control formulation explicitly, but it has several problems: the method is computationally expensive for large-scale recurrent networks and the control is unstable for recurrent networks with chaotic dynamics since a gradient collection through time diverges due to the chaotic instability.

In this paper, we propose another method in order to reduce the computational cost and realize a "stable" control for recurrent networks with chaotic dynamics. Firstly we investigate relations between the complexity and parameters of the network configuration under a restriction. The new method is based on the relation which allows us to approximate the gradient collection in a fashion without time evolution. Simulation results show that new method can control the exponent for recurrent networks with chaotic dynamics.

2 Complexity of recurrent networks

A subset of the Lyapunov exponents is used as a measure of the dynamical complexity. The complexity is defined strictly by using the complete set of the exponents. Calculation of the complete set is, however, computationally expensive. In the following, only the largest Lyapunov exponent will be concerned since it can be decided whether the systems are chaotic or not by using the largest exponent: if a system is chaotic then the largest exponent λ is greater than 0, otherwise the exponent is less than 0.

Fully connected recurrent networks composed of N units are considered. Activate functions of neurons are sigmoid. Letting t be discrete time, $t = 1, 2, \cdots$, the outputs of neurons x_i are governed by the following difference equations:

$$x_i(t+1) = \frac{1}{1+\exp(-2s_i(t+1)/s_0)}, \quad (1)$$

$$s_i(t+1) = \sum_{j=1}^{N} w_{ij}x_j(t) + \theta_i, \quad (2)$$

$i = 1, 2, \cdots, N$. s_i are inputs, w_{ij} connection strengths, s_0

gain coefficients of the sigmoid functions and θ_i biases. In this case, the largest Lyapunov exponent λ is calculated by the following equations[2]

$$\lambda = \lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\log L(t+1). \quad (3)$$

Where $L(t)$ denotes a Euclidean norm $\|L(t)\|$ and elements of the vector $L(t+1) = [L_1(t+1), L_2(t+1), \cdots, L_N(t+1)]^\top$ are given by

$$L_i(t+1) = \frac{2}{s_0}X_i(t+1)\sum_{j=1}^{N} w_{ij}\delta L_j(t), \quad (4)$$

where $\delta L = [\delta L_1(t), \delta L_2(t), \cdots, \delta L_N(t),]^\top$ is given by

$$\delta L(t) = \frac{L(t)}{\|L(t)\|}, \quad (5)$$

and $X_i(t)$ denotes $x_i(t)(1-x_i(t))$ and $\delta L(0)$ is supposed by an unit vector. This implies you can calculate the largest Lyapunov exponent λ using time series of the networks, $x(0), x(1), \cdots, x(T)$ and the weights w_{ij}.

3 Control of the complexity

3.1 Gradient method

How do you set the Lyapunov exponent of the networks to the desired value? As mentioned in the previous section, the exponent λ is a function of the network parameters such as w_{ij}. That is, the question is equivalent to how you design the parameters which generate dynamics with the desired exponent. One of the methods to achieve this design is a gradient method presented in our previous paper[2]. In the following, the gradient method is described briefly.

Letting λ^{obj} be the desired exponent, the change of parameter δw_{ij} is given by

$$\begin{aligned}\delta w_{ij} &= -\eta\frac{\partial e_\lambda}{\partial w_{ij}},\\ &= -\eta(\lambda-\lambda^{obj})\frac{\partial\lambda}{\partial w_{ij}},\end{aligned} \quad (6)$$

where $e_\lambda = (\lambda-\lambda^{obj})^2/2$ is a squared error and η is a positive coefficient. If the gradient $\partial\lambda/\partial w_{ij}$ is obtained, the change of parameter δw_{ij} is able to be calculated.

Our previous control method of λ needs to calculate $\partial\lambda/\partial w_{ij}$ with a gradient collection through time. This collection through time requires numerous computational resources and it makes the control be unstable for networks with chaotic dynamics.

3.2 Approximation method

To solve above problems, a qualitative method based on an approximate relation between the complexity and parameters of the network configuration has been proposed previously[3]. The method is as follows: At the first we give a restriction with respect to the network configuration which allows us to introduce a key parameter $\overline{\sigma^2}$. We initialize w_{ij} randomly, then define the biases θ_i by

$$\theta_i = -\frac{1}{2}\sum_j w_{ij}. \quad (7)$$

For large-scale networks with above configuration, a variance $\overline{\sigma^2}$ of the inputs of neurons is calculated stochastically as[3]

$$\overline{\sigma^2} = \frac{1}{12N}\sum_{i,j} w_{ij}^2. \quad (8)$$

From above arrangements, you have

$$\frac{\partial\lambda}{\partial w_{ij}} = \frac{\partial\lambda}{\partial\overline{\sigma^2}}\cdot\frac{\partial\overline{\sigma^2}}{\partial w_{ij}}. \quad (9)$$

$\partial\overline{\sigma^2}/\partial w_{ij}$ is calculated by (8) as

$$\frac{\partial\overline{\sigma^2}}{\partial w_{ij}} = \frac{1}{6N}w_{ij}. \quad (10)$$

$\partial\lambda/\partial\overline{\sigma^2}$ is calculated by a qualitative relation between the λ and $\overline{\sigma^2}$ as

$$\frac{\partial\lambda}{\partial\overline{\sigma^2}} \approx \begin{cases} A_1A_2\exp(-A_2\overline{\sigma^2}) & ,(\lambda<0),\\ B_1 & ,(\lambda>0).\end{cases} \quad (11)$$

The approximate gradient $\partial\lambda/\partial w_{ij}$ is then calculated by (9) $\sim$ (11) where A_1, A_2, B_1 are positive constants. The method is practical for large-scale networks since the method requires only $O(N^2)$ run time to control the exponent λ. However the method was based on experimental results rather than theoretical ground, thus the control isn't always stable.

3.3 Stochastic analysis

In this paper we propose a new approximation method by analyzing the relation stochastically. The main point of the following analysis is to approximate the collection through time of the length $\log L(t+1)$ in (3) to a fashion without time evolution since a collection through time for chaotic systems results in computational divergence due to chaotic instability. We try to get the average length $\overline{\log L}$ of the $\log L(t+1)$ through time. If $\overline{\log L}$ is obtained, (3) is converted to the following:

$$\lambda = \overline{\log L}. \quad (12)$$

Our strategy to approximate $\overline{\log L}$ is to calculate the expectation of $\log L_i(t+1)$ as a function of the key parameter $\overline{\sigma^2}$[4].

First, let's suppose elements of λ, for instance, $x_i(t)$, w_{ij}, $i,j = 1,2,\cdots,N$, $t = 1,2,\cdots,T$ are independent of each other in order to calculate the expectation more simply. For large-scale recurrent network with above supposition, the expectation of λ is calculated by the law of large numbers and (4), (12) as

$$\begin{aligned}\lambda &\approx \frac{1}{2}\Big\{\log N + \log E\big[\{L_i(t+1)\}^2\big]\Big\},\\ &\approx \frac{1}{2}\Bigg\{\log\frac{4N}{s_0^2} + \log E\big[\{X_i(t+1)\}^2\big]\\ &\qquad + \log E\left[\left\{\sum_{j=1}^{N} w_{ij}\delta L_j(t)\right\}^2\right]\Bigg\}.\end{aligned} \quad (13)$$

Second, $E\big[\{\sum_{j=1}^{N} w_{ij}\delta L_j(t)\}^2\big]$, $E\big[\{X_i(t+1)\}^2\big]$ are calculated in order to obtain the expectation of λ. Where $E[z]$ is the expectation of z. Letting w_{ij} be defined uniformly, $E\big[\{\sum_{j=1}^{N} w_{ij}\delta L_j(t)\}^2\big]$ is calculated by the law of large numbers as

$$\begin{aligned}E\left[\left\{\sum_{j=1}^{N} w_{ij}\delta L_j(t)\right\}^2\right] &\approx E\left[\sum_{j=1}^{N}\{w_{ij}\delta L_j(t)\}^2\right],\\ &\approx N\cdot E\left[w_{ij}^2\right]\cdot E\big[\{\delta L_j(t)\}^2\big].\end{aligned} \quad (14)$$

$E\left[w_{ij}^2\right]$ is calculated by (8) as

$$E\left[w_{ij}^2\right] = \frac{1}{N^2}\sum_{i,j} w_{ij}^2, = \frac{12}{N}\overline{\sigma^2}. \quad (15)$$

$E\left[\{\delta L_j(t)\}^2\right]$ is calculated by the law of large numbers and (5) as

$$E\left[\{\delta L_j(t)\}^2\right] = \frac{1}{N}\sum_{j=1}^{N}\{\delta L_j(t)\}^2, = \frac{1}{N}. \quad (16)$$

Thus the expectation of λ is calculate by (13) $\sim$ (16) as

$$\lambda \approx \frac{1}{2}\log\left(48\frac{\overline{\sigma^2}}{s_0^2}E[\{X_i(t+1)\}^2]\right). \quad (17)$$

Third, we try to get $E[\{X_i(t+1)\}^2]$. From (1) the sigmoid function is given on a power series representation as follows [10].

$$x_i = 1\Big/\left\{1+\sum_{n=0}^{\infty}\frac{(-1)^n\cdot 2^n}{n!}\cdot\left(\frac{s_i}{s_0}\right)^n\right\}, = \sum_{n=0}^{\infty}\mathrm{F_1}(n)\cdot\left(\frac{s_i}{s_0}\right)^n,$$

$$\mathrm{F_1}(n) \equiv \begin{cases} 1/2, & (n=0), \\ \sum_{k=1}^{n}\frac{(-1)^{k+1}\cdot 2^{k-1}}{k!}\cdot\mathrm{F_1}(n-k), & (otherwise). \end{cases} \quad (18)$$

Where s_i, x_i are defined as $s_i(t), x_i(t)$. From (18) x_i^m, $m = 2, 3, \cdots$, are given by

$$x_i^m = \sum_{n=0}^{\infty}\mathrm{F_m}(n)\cdot\left(\frac{s_i}{s_0}\right)^n,$$

$$\mathrm{F_m}(n) \equiv \sum_{k=0}^{n}\mathrm{F_{m-1}}(k)\cdot\mathrm{F_{m-1}}(n-k). \quad (19)$$

Thus $\left\{X_i(t+1)\right\}^2$ can be approximated on a power series representation as

$$\left\{X_i(t+1)\right\}^2 \approx \begin{cases} \sum_{n=0}^{M_1}\mathrm{H}(n)\cdot\left(\frac{s_i}{s_0}\right)^n, & \left(\left|\frac{s_i}{s_0}\right|<\alpha\right), \\ 0, & (otherwise), \end{cases}$$

$$\mathrm{H}(n) \equiv \{\mathrm{F_2}(n) - 2\mathrm{F_3}(n) + \mathrm{F_4}(n)\}. \quad (20)$$

Where M_1 is a suitable natural number and α is positive constant. The probability density $g(x, 0, \overline{\sigma^2})$ of the expectation of the input $s_i(t)$ can be given on a power series representation as follows since the probability density is defined as normal distribution [3][10].

$$\mathrm{g}(x,0,\overline{\sigma^2}) = \frac{1}{\sqrt{2\pi\overline{\sigma^2}}}\exp\left(-\frac{x^2}{2\overline{\sigma^2}}\right), = \begin{cases} \sum_{n=0}^{M_2}\frac{\mathrm{R}(n)}{\sqrt{\overline{\sigma^2}}}\cdot\left(\frac{x}{\sqrt{\overline{\sigma^2}}}\right)^{2n+1}, & \left(\left|\frac{x}{\sqrt{\overline{\sigma^2}}}\right|<\beta\right), \\ 0, & (otherwise), \end{cases}$$

$$\mathrm{R}(n) \equiv \frac{(-1)^n}{\sqrt{2\pi}\cdot 2^n\cdot n!}.$$

(21)

Where M_2 is a suitable natural number and β is positive constant. Thus $E[\{X_i(t+1)\}^2]$ can be given on a power series representation as

$$E[\{X_i(t+1)\}^2] = \int_{-\infty}^{\infty}\{X_i(t+1)\}^2\cdot\mathrm{g}(x,0,\overline{\sigma^2})\,dx, = \sum_{\substack{0\le n\le M_2\\ 0\le k\le n}}\frac{\mathrm{Q}(n,k)\cdot t^{2n+1}}{s_0^{2k}\cdot\left(\sqrt{\overline{\sigma^2}}\right)^{2(n-k)+1}} + \sum_{\substack{M_2+1\le n\le 2M_2\\ n-M_2\le k\le M_2}}\frac{\mathrm{Q}(n,k)\cdot t^{2n+1}}{s_0^{2k}\cdot\left(\sqrt{\overline{\sigma^2}}\right)^{2(n-k)+1}},$$

$$\mathrm{Q}(n,k) \equiv \gamma\cdot\frac{2\cdot\mathrm{H}(2k)\cdot\mathrm{R}(n-k)}{(2n+1)},$$

$$t \equiv \min(\,\alpha s_0,\ \beta\sqrt{\overline{\sigma^2}}\,). \quad (22)$$

Where we suppose $M_1 = 2M_2$ and γ is positive constant. Finally from (17), (22) you can have the new approximate relation between λ and $\overline{\sigma^2}$.

Fig. 1 shows λ as functions of the parameter $\overline{\sigma^2}$. In this case, the networks are composed of 100 neurons, s_0 was 0.2 and T was 100. Note that our new approximation method by deciding experimentally M_2 was 5 and α was 2 and β was 2 and γ was 1.2 can predict the approximate relation between λ and $\overline{\sigma^2}$. In addition, from the same simulation for different-scale networks we find that these parameters such as M_2, α, β and γ are independent of the scale of the network, if the networks are composed of large number of neurons enough to approximate stochastic relations.

3.4 Proposed complexity control method

From above results, $\partial\lambda/\partial\overline{\sigma^2}$ is calculated by (17) as

$$\frac{\partial\lambda}{\partial\overline{\sigma^2}} = \frac{1}{2}\left\{\frac{1}{\overline{\sigma^2}} + \frac{1}{E[\{X_i(t+1)\}^2]}\cdot\frac{\partial E[\{X_i(t+1)\}^2]}{\partial\overline{\sigma^2}}\right\}. \quad (23)$$

Where $\partial E[\{X_i(t+1)\}^2]/\partial\overline{\sigma^2}$ can be calculated by (22). Thus the partial differential coefficients $\partial\lambda/\partial w_{ij}$ without the collection through time is calculated by substituting (23) into (9).

4 Simulation results

Our control methods have been tested on a design task which requires the fully connected networks to have a de-

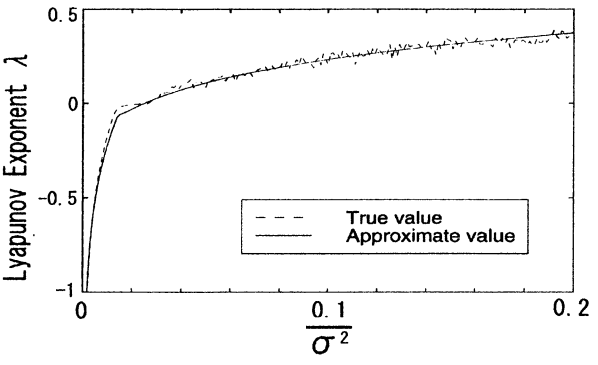


Figure 1: The Lyapunov exponent λ as functions of the parameter $\overline{\sigma^2}$.

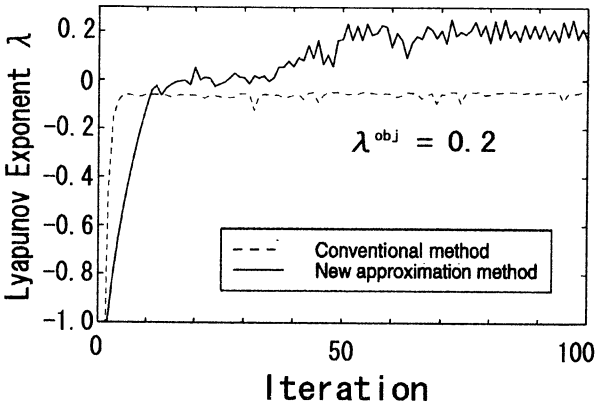


Figure 2: The Lyapunov exponents λ as functions of the iteration by a conventional method and the proposed one ($N = 20$, $\lambda^{obj} = 0.2$).

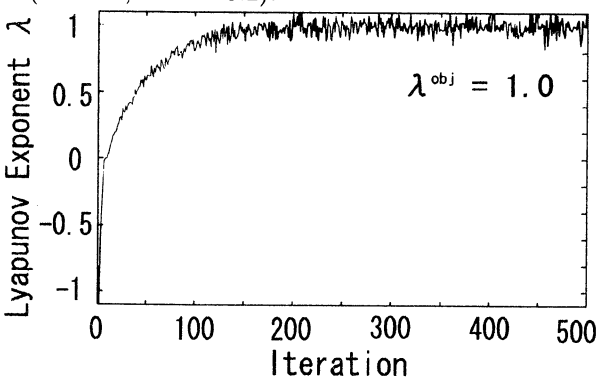


Figure 3: The Lyapunov exponent λ as a function of the iteration by the proposed approximation method ($N = 100$, $\lambda^{obj} = 1.0$).

sired value of the largest Lyapunov exponent. In this task, the connection weights of the network were initialized randomly, then changed by our methods.

Fig. 2 shows the exponent λ is controlled to the value of the chaotic dynamics by our methods. The network controlled was composed of 20 neurons and λ^{obj} was 0.2 and M_2 was 5 and α was 1.2 and β was 2 and γ was 1.2 and s_0 was 0.2 and T was 100. Note that the exponent λ converges to the desired value by our new approximation method, while the conventional one fails to control the exponent λ around $\lambda \approx 0$ because of the chaotic instability. In this case our new approximation method requires only 0.032 seconds' cpu-time to calculate the partial differential coefficients $\partial\lambda/\partial w_{ij}$ while the conventional one requires 986 seconds' cpu-time. This implies the conventional method isn't practical for large-scale networks in comparison with our new approximation one. Fig. 3 shows similar simulation result by our new approximation method for large-scale networks. The network controlled was composed of 100 neurons and λ^{obj} was 1.0. Note that the exponent λ converges to the desired value. In this case the cpu-time is 0.094 seconds. Thus our new approximation method is practical for large-scale networks.

5 Conclusions

In this paper we analyzed a stochastic relation between the largest Lyapunov exponent and network parameters for large-scale fully connected recurrent networks with asymmetric connection weights w_{ij} and biases $\theta_i = -\frac{1}{2}\sum_j w_{ij}$. The relation allows us to approximate the gradient collection through time in a fashion without time evolution.

Simulation results show that effectiveness of proposed method with respect to the computational cost and the stable control of the Lyapunov exponent of recurrent networks with chaotic dynamics.

References

[1] G. Deco and B. Schürmann, "Dynamic modelling chaotic time series," *Computational Learning Theory and Neural Learning Systems*, Vol.4 of Making learning Systems Practical, MIT Press, Cambridge, Mass., chapter 9, pp.137-153, 1997

[2] N. Honma, M. Sakai, K. Abe and H. Takeda, "Complexity Control Methods of Dynamics in Recurrent Neural Networks(in Japanese)," *Trans. SICE*, Vol.35, No.1, pp.138-143, 1999

[3] N. Honma, K. Kitagawa and K. Abe, "Effect of complexity on learning ability of recurrent neural networks," *Proc. of AROB*, Vol.2, pp.97-101, 1998

[4] N. Honma, K. Kitagawa, K. Abe and H. Takeda, "An Autonomous Criterion of Learning Methods for Recurrent Neural Networks," *Proc. of the 2nd ASCC*, Vol.2, pp.219-222, 1997

[5] B. Irie and S. Miyake, "Capabilities of three-layered perceptrons," *Proc. of IEEE ICNN 88* , pp.I-641, 1988

[6] M. Jordan, "Generic Constraints on Underspecified Target Trajectories," *Proc. International Joint Conference on Neural Networks(IJCNN-89)*, Vol.1, pp.217-225, 1989

[7] K. S. Narendra and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Trans. Neural Networks*, Vol.1, No.1, pp.4-27, 1990

[8] J. C. Principle, A. Rathie and J. Kuo, "Prediction of chaotic time series with neural networks and the issue of dynamic modeling," *International Journal of Bifurcation and Chaos* , Vol.2, No.4, pp.989-996, 1992

[9] J. C. Principle and J. Kuo, "Dynamic modelling chaotic time series with neural networks," In G. Tesauro et al.(Eds.), Neural Information Processing System 7, pp.311-318, 1995

[10] M .Ueda, Y .Okada and Y .Yoshitani, Probability and statistics(in Japanese), Dainippon Tosho Co.,Ltd., Tokyo, 1980

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

SPEECH RECOGNITION USING NEURAL NETWORKS

• P.N.GIRIJA and P.SAMBASIVA RAO
DEPARTMENT OF COMPUTER / INFORMATION SCIENCES
ARTIFICIAL INTELLIGENCE LAB
UNIVERSITY OF HYDERABAD
HYDERABAD - 500 046 ,A.P., INDIA.
E-mail : pngcs@uohyd.ernet.in

ABSTRACT

Speech Recognition is the process that allows one to communicate with a computer by speaking to it. In this paper a neural network model to recognize isolated words from the sampled speech signal is presented. It consists of a Kohonen Map and a Perceptron. The novel idea in this system is the usage of a Kohonen Map as these feature extractor, which converts phonetic similarities of speech frames into spatial adjacency in the map This property simplifies the classification task. The input to this map is Cepstral Coefficients which are obtained from Linear Predictive Coefficients (LPCs). The output of Kohonen Map is given as input to Perceptron. The system performance is evaluated for recognition of a limited number of TELUGU digits (numbers "One" through "Ten"). TMS320 based TI-speech system is used o to carry out this work.

Key Words: Linear Predictive Coefficients, Cepstralralal Coefficients, Speech Recognition , Kohonen Map,Perceptron.

1. INTRODUCTION

Automatic Speech Recognition (ASR) is often justified on the grounds that it offers a *"third hand"* or allows *"eyes free"* operation **[1].**

1.1 Difficulties in Speech Recognition

Work load and task stress significantly decreases recognition performance **[2,3].** Due to the presence of background noise and channel mismatches the recognition accuracy decreases and it varies between male and female speakers **[4,5].**

1.2 Approaches to Speech Recognition

Approaches for robust speech recognition can be summarized as 1) Better training methods 2) Improved front-end processing 3) improved back-end processing or robust speech recognition measures **[2,3,5].** Neural networks have very promising features such as the abilities of learning and generalization and they do not make assumptions about their inputs. Most of the algorithms are based on perceptron **[6].** Perceptron attractiveness is due to it's well known learning algorithm known as **Back propagation algorithm [7]** but with increasing number of connections the training time increases and also makes it more probable to fall in a poor local minimum basin of attraction **[8].**

2. PRE-PROCESSING

2.1 Feature Extraction methods

Each speech signal contains information about the text and the speaker. Since speech signal occupies more space the same can be represented more efficiently as parameters and the parametric representation is a slowly varying function than the speech signal. Speech parameters are extracted from the speech signal using signal processing techniques **[9 - 12].**

3. FEARURE EXTRACTION WITH KOHONEN MAP

3.1Introduction

Self-organizing feature maps are based on competitive learning. The output neurons of the network compete among themselves to be activated or fired, with the result that only one output neuron or one neuron per group, is on at any time **[13 - 16].**

3.2 Application of the map to speech recognition

Kohonen map can be used as a vector quantizer. This algorithm can perform relatively well in noise because the number of classes is fixed, weights adapt slowly, and adaptation stops after training. The training set parameter vectors are obtained from natural speech, and are reordered randomly before being used for training. The final outputs of the Kohonen map are just two integers, which help the perceptron for better classification. The perceptron has been considered as final classifier **[7,8].**

4. DESIGN

The speech recognition system consists of the modules like 1.Preprocessing 2.Feature Extraction with Kohonen

map 3. Perceptron as the final classifier. Flow chart of the speech recognition system is shown in Fig.4.1(a).

4.1 Preprocessing

The speech signal is digitized at a sampling rate of 8000 Hz. Each utterance of the Isolated Word is end point detected and pre-emphasized with a coefficient of 0.95. The speech is then segmented into a sequence of frames **[18-21].** Each frame is processed by a Hamming window and analyzed by using the 12-th order model based on the autocorrelation method **[22].**. The block diagram of the Preprocessing unit is shown in **figure 4. 1 (b) .**

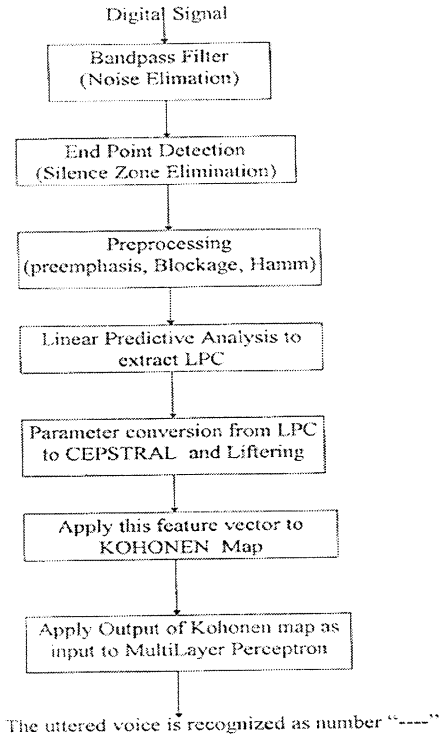


Fig4.1(a) low chart for Speech Recognition System

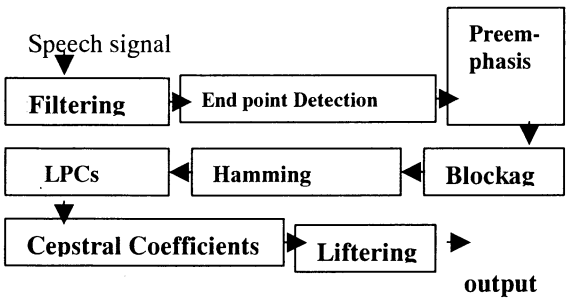


Fig.4.1(b) Block diagram of preprocessing unit

4.2 Feature extraction with kohonen map

The training set parameters are obtained from natural speech, and reordered randomly before being used for the training **[17]**. The final outputs of the Kohonen map, which are used as features, are the indices of the winner neuron for each parameter vector.

4.3 Perceptron as the final classifier

The final stage of the recognition system is a perceptron which acts as a nonlinear classifier. There are two hidden layers. The first hidden layer consists of eighteen, and the second of fourteen neurons. Each of the output layer's neurons corresponding to one of TELUGU numbers 'One' to 'Ten'.

4. 4 Training / Recognition

The Speech Recognition System can be grouped into two tasks namely1. Training. 2. Recognition. In order to train the system a database consisting of 2000 words uttered by seventy male and thirty female speakers is collected. Each one of the specified words were uttered twice by each speaker. These 2000 words are preprocessed and applied to Kohonen Map. The output of this Kohonen Map is added with the desired outputs to give as input to the Multilayered Perceptron. Now we trained this Network till error reaches a reasonable range.

The second task, *Recognition,* is to recognize any word (i.e. the vector, which is not having the desired outputs) in the database .A test feature vector is constructed from the speech signal, and processed through the Kohonen and MLP networks then the uttered voice is recognized as the number which is having highest value.

5.IMPLEMENTATION

For noise reduction the band pass filter is used and end points **[13]** are detected before feature extraction.

5.1 Feature Extraction

The LPC Coefficients and Cepstral Coefficients are extracted from the speech signal.

5.2 Implementation of Self Organizing Map (SOM)

The SOM is a relatively uncomplicated network in that it has only two layers of units. But the output layer is a two-dimensional matrix, rather than as a simple one-dimensional vector. To accommodate this new dimension, the matrix will be decomposed conceptually into a single vector containing all the row vectors from the original matrix.

There are two aspects of training process that are relatively simple to implement, that are needed to initialize the SOM and to apply all input vectors (set-input) to the input layer of the network. During the recognition phase the given test signal is process-ed for extracting the same parameters , as extracted during the training phase and a similar feature vector is trained. Now apply the feature vector to the Kohonen map. The output of this map is a vector consisted of all the indices of the winner coordinates. This vector is the input to the Multilayer Perceptron (Back propagation) network. The output of this Perceptron is the node, which has highest output among all the ten output nodes. The system recognizes the test signal as the number of that node which has highest value.

5.3 Experimental Results

For training the neural network a database consisting of 2000 words uttered by seventy male and thirty female speakers is used . Each one of the ten specified words (**Table 5.1**) are uttered twice by each speaker. For testing another database consisting of 1000 words is used.

5.4 CONCLUSIONS

It is observed that the percentage of accuracy increases whereas percentage of error decreases with the increasing number of speakers, number of nodes. It is also observed that the percentage of recognition accuracy is more for males compared to female speakers. The benefits of this system are speed and simplicity.

5.5 SCOPE OF THE FUTURE WORK

The system can be made to increase the number of words, whether the spoken word is in the set or out of the set and the efficiency of the system can be improved by trying with various architectures of the Perception Classifier.

Number	In Telugu
1	Okati
2	Rendu
3	Moodu
4	Naalugu
5	Aidu
6	Aaru
7	Aedu
8	Enimidi
9	Tommidi
10	Padi

T able 5.1(a) Ten Telugu words

Sex	Female	male
1	73.3	60
2	83.3	63.5
3	83.3	66.4
4	83.3	29.3
5	86.6	75.7
6	76.6	50
7	93.3	80
8	80	78.5
9	75	77.2
10	61.6	78.5

Table 5.1(b)Accuracy Percentages of the recognizer

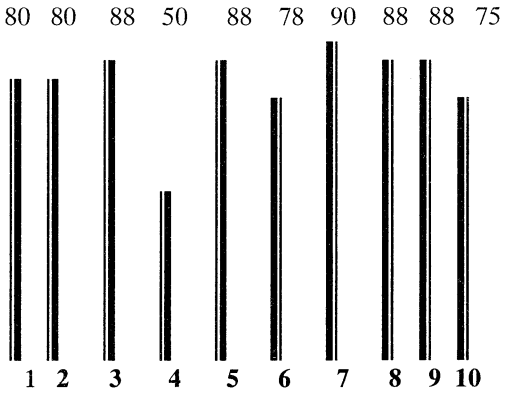


Fig. 6.1 Percentage of recognition accuracy tested by ten male speakers (Y-axis –Percentage of accuracy)

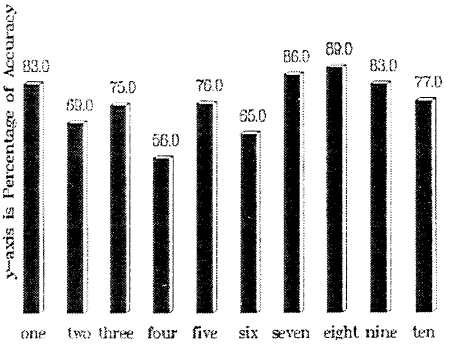
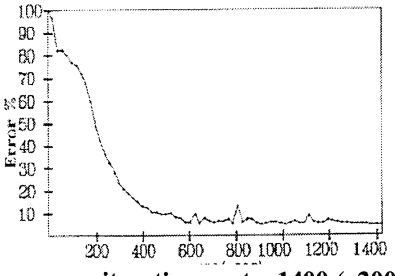
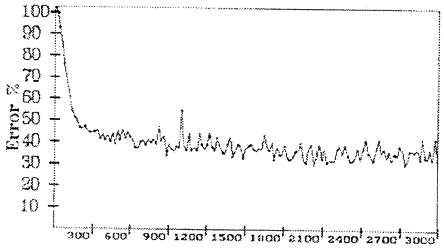


Fig. 6.2 Percentage of recognition accuracy tested by female speakers



iterations upto 1400 (x200)

Fi g. 6.3. percentage of error curve during Training of ten speakers



Iterations upto 3000(x1000)

Fig: 6. 4 Percentage of error curve during training of Fifty Speakers

REFERENCES

[1] Baber C [1], Mellor B[2], Graham R [3], Noyes and Tunley J.M. [4], (1990) ,Work load and the use of Automatic Speech Recognition, Speech Communication, Vol . 20, pp. 37- 53.

[2] Chen Y (1988), Cepstral domain stress compensation for robust Speech Recognition, IEEE Trans. on ASSP Vol. 36, pp. 433-439.

[3] Hansen J H L. (1989), Evaluation of Acoustic correlates of Speech under stress for robust Speech Recognition, IEEE Proc. 15th Northeast Bioengineering Conf., Boston, MA, pp. 31-32.

[4] Hansen J H L (1996), Analysis and compensation of speech under stress and noise for environmental robustness in Speech Recognition, Speech Communication, Vol 20, pp 151-173.

[5] Hansen J H L and Bria O N (1990), Lombard effect compensation for robust automatic Speech Recognition in noise, ICSLP-90: Int. Conf. on Spoken language processing, Kobe, Japan. pp. 1125-1128.

[6] Wasserman P.D (1989), Neural Computing Theory and Practice, VANNOSTRAND Reinhold, New York .

[7] Rumelhart DE, Hinton GE and Williams R J (1986) Leaming Internal Representations by Error Propagation in Parallel Distributed Processing , Vol. 1, D.E. Rumelhart, J.L McClellad, Eds. Cambridge, MA., MIT Press.

[8] Hush D R and Horne B G (1993) , Progress in Supervised Neural Networks, IEEE SP Magazine, pp: 8-39.

[9] Schafer R W and Rabiner L R (1974), Parameter Representations of Speech, Speech Recognition invited Papers presented at the IEEE Symposium, Edited by D. Raj Reddy, Academic Press, Inc.,pp. 99-150.

[10] Rabiner L R and Schafer R W (1978) , Digital Processing of Speech Signals, Prentice-Hall Inc., Eaglewood Cliffs, New Jersey.

[11] Barnwell T P, Nayebi, K and ,.Richardson C H (1996) , Speech Coding *A Laboratory Textbook",* John Wiley & Sons, Inc.

[12] O'Shaughnessy D (1987) , "Speech Communication , Human and and Machine , Addition-Wesley Publishing Company.

[13] Kohonen T (1990), The Self Organizing Map, Proc. of the IEEE, Vol. 78, No. 9, pp: 1464-1476.

[14] Kokkenien M and Toryjkola K (1990), using self organizing maps and multi layered feed-forward nets to obtain phonetic transcriptions of spoken utterances, Speech Communication, p.p 541-549.

[15] Lippman R P (1987) , An Introduction to Computing with Neural Nets, ASSP, pp: 4-22.

[16] Symon Haykins (1996), Self-Organizing Competitive Leaming, in Neural Networks, IEEE Press.

[17] Taatabaee V [1], Azimisdjadi B [2], .Zahirazami S B [3] , and Lucas C [4] (1994) , Isolated Word Recognition Using a Hybrid Neural Network, ICAS SP.

[18] Lin C H [1]., Wu C H [2], Ting P Y [3] and Wang H M [4] (1996), Frame works for recognition of mandarin syllables and tones using sub-syllabic units, Speech Communication, Vol. 18, pp. 175-190.

[19] Lee K F (1990), Context-dependent phonetic hidden markov models for speaker independent continuous speech recognition, IEEE Trans. on ASSP, Vol. 38, No 4, pp. 599-609.

[20] Lee C H et al (1990), Acoustic modeling of sub word units for speech recognition, Computer Speech and language, Vol. 4, pp. 127-165.

[21] Lee C.H et al (1992), Improved Acoustic modeling for large vocabulary continuous Speech Recognition, Computer Speech and language, Vol. 6, pp. 103-127.

[22] Markel J D and Gray A H (1976), Linear Prediction of speech, Springer-Verlag.

[23] Tokhura Y (1987), A weighted cepstral distance measure for speech recognition, IEEE Trans on ASSP, Vol. ASSP- 5,No 10, p.p 1414-1422.

[24] Makhoul J , Roucos S and Gish H (1985), Vector quantization in speech coding, Proc. of the IEEE,vol. 73, No. 11.

[25] Schroder M R (1981) , Direct (Non Recursive) relations between cepstrum and predictive coefficients, IEEE Trans. on ASSP , Vol.ASSP-29,No. 2,p.p 297-301.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Learning-possibility that neuron model can recognize rotation in three-dimension

Yasuhiro SEKIYA†, Qianyi WANG†, Tomoo AOYAMA†, Hiroki TAMURA††, Zheng TANG†††

† Faculty of Engineering, Miyazaki University., 1-1, Gakuenkibanadai-nishi, Miyazaki-shi, 889-2192, Japan.
(E-mail: tgb013u@cc.miyazaki-u.ac.jp)
†† Asahi Engineering, Co. Ltd., 1-8, Kohnan 4 chome, Minato-ku, Tokyo, 108-0075, Japan.
††† Faculty of Engineering, Toyama University., 3190, Gofuku, Toyama-shi, 930-8555, Japan.

Abstract

We propose a neuron model which is possible to learn rotation movement in three-dimensional. The neuron model imitating structure of a neuron has the system resemble a neuron. We considered a neuron system based on the arguments, and wanted to examine whether the system had reasonable function. Koch, Poggio and Torre believed that inhibition signal would shunt excitation signal on the dendrites. They believed that signal functions as input and delayed input. Thus, they were sure that the function for directional selectivity was arisen in the delay system. Koch's concept is important; so, we construct the neuron system with their concept. We initialized the connections and the dendrites by random data, and trained them by the back-propagation algorithm for three-dimensional movement. The neuron model after learning to the function for directional selectivity can perceive depth-rotation. The neuron model after learning is similar to the real neuron's morphology. We made sure the detection of three-dimensional movement in the system.

keywords : *MST area, synapse strength, dendritic, three-dimensional exercises, nonlinear interaction*

1. Introduction

The visual recognition is very important. We wish to understand visual function of a creature's visual information processing. Recently, the research of a monkey's brain is remarkably advanced. In brain cortex of monkey, it is reported that approximately 55 % is occupied in the visual area[1]. The visual area exists in six areas, and it is composed of *Visual area 1st~5th*, and *6th area* named by *Medical Superior Temporal* (MST) *area*[1]. In MST area, it is guessed that MST area recognize exercise of stereopsis. In monkey's MST area, it is known that existence of neurons responding with large planar motion and optical-flows of radiation or rotation around a point[2]. Gibson proposed conception of "the optical-flow"[3]. "the optical-flow" means the following description; the movement by the object of external is perceived, and the image projected on retia produces flow. He believed that animals perceived three-dimensional motion to be the optical-flow. Thus, Monkey perceived directional selectivity in the visual area of brain.

C.Koch, T.Poggio and V.Torre found directionally selective cells in the retina of cat. They suggested that there may be 4 types of retinal ganglion cell in cat. Their analysis also showed that γ and δ cells may have strong interactions between excitation and inhibition of the shunting type, and that δ cell's morphology is almost ideal for the mechanism of directional selectivity[4].

Thus, we considered whether inhibition of the shunting type is performed in MST area by C.Kochs' conception. In this paper, we made the shunting function apply to our neuron model. Then, we studied whether a neuron model is perceived rotation movement in space. The neuron model after learning is similar to the real neuron's morphology. Therefore, it is considered that the neuron model is close to real neuron and give a good modeling for the MST area.

2. Koch, Poggio and Torre model

C.Koch, T.Poggio and V.Torre found directionally selective cells in the retinae of cat. Figure 1 is a model of movement detection with guessed δ cells. This scheme for directional selectivity to motion is based on delay system on a ganglion cell. Excitation and inhibition from the photoreceptors to the ganglion cell dendrites are carried through fast and slow pathway, respectively. Figure 1 shows that in perspective, δ-like dendrite receives topographic inputs from photoreceptors. The connections shown by broken lines represent delayed input. When a continuous signal is inputted with the order of 1→2→3→4→5→6 in photoreceptors, because the delayed inhibition shunts the route of input, the movement detection isn't performed. When a continuous signal is inputted with the order of 6→5→4→3→2→1 in photoreceptors, because

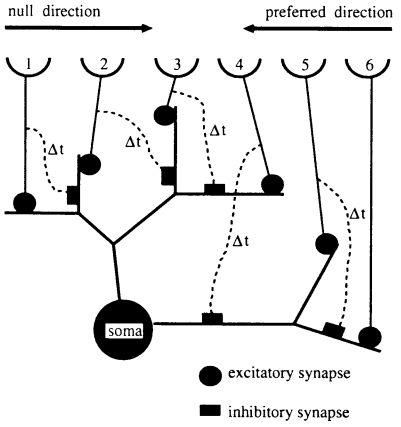


Figure 1: Mechanism of directonal selectivity.

the delayed inhibition doesn't disturb route of input, movement detection is performed. Its properties appear that directionally selective cells may have a δ-like morphology[4].

3. Neuron model

In this paper, we propose a neuron model with the dendritic of neuron[5]. We considered that the neuron model has 4-layers.

Synapse Layer: The function in this layer is the synapse strength function and yields synapse interaction by *excitatory postsynaptic potential*(EPSP) and *inhibitory postsynaptic potential*(IPSP) and *resting membrance potential*(RMP). The synapse strength is generated in the layer, when a learning process is executed. The synapse strength function is given by

$$Y_{ij} = \sigma_{ij} X_{ij} + \gamma_{ij} \bar{X}_{ij} \tag{1}$$

where X is inputs, σ is a strength parameter, and γ is a rest parameter.

AND Layer: This layer operates as dendrites, whose function simulates the nonlinear interactions between excitatory and inhibitory synapses. The function in the second layer resembles the AND operation. The AND operation of the input variable yields the value near 1 or 0 if all input variables are near 1 or 0. And, the AND operation of the input variable yields the same value if all input variables are same. For any n-variable logic function, only 2^{n-1} AND gates will be necessary. In order to do learning with gradient descent, derivatives of the AND function are needed. However, the derivatives of the AND function do not exist. Here, we use a soft-minimun operator. And, the mean of Y_{ij} is added to realize figure 2.

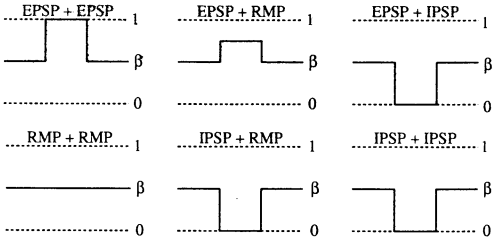


Figure 2: Interaction of synapse.

$$AND_j = \frac{\sum_{i=1}^{n} Y_{ij} e^{-uY_{ij}}}{\sum_{i=1}^{n} e^{-uY_{ij}}} \cdot \frac{\sum_{i=1}^{n} Y_{ij}}{n} \tag{2}$$

where u is a positive constant.

OR Layer: This layer performs function like the OR operation. The OR operation yields near 1 whenever at least one of the variables is near 1, and near 0 whenever all of the variables is near 0, where each $x_1, x_2, \cdots, x_n$ assumes the value near 0 or 1. For learning purposes, we use a soft-maximun operator:

$$OR = \frac{\sum_{j=1}^{m} AND_j e^{vAND_j}}{\sum_{j=1}^{m} e^{vAND_j}} \tag{3}$$

where v is a positive constant.

Neuron Layer: The Neuron-layer functions as a soma in the neuron. This layer performs the Sigmoid operation. The neuron yields the value 0 or 1 when the output value from the OR-layer is above or below a certain threshold. Thus, this layer functions as AD converter. For learning purposes, we use a sigmoid operator:

$$O = \frac{1}{1 + e^{-k(wOR - \theta)}} \tag{4}$$

where w, θ are connection parameters and k is a positive constant.

The neuron model mentioned above is a feed-forward network with continuous functions. Thus, the error back-propagation algorithm will be valid for the neuron model. A learning rule for the neuron model can be readily derived from the condition of least squared error between the actual output O and the desired output T(Teacher's signal) defined as

$$E_p = \frac{1}{2}(T_p - O_p)^2 \tag{5}$$

Since the minimization of the error requires the strength, weight and threshold changes to be in the negative gradient direction, we take

$$\Delta\sigma_{p,ij} = -\eta\frac{\partial E_p}{\partial\sigma_{p,ij}} \quad (6)$$

$$\Delta w_p = -\eta\frac{\partial E_p}{\partial w_p} \quad (7)$$

$$\Delta\theta_p = -\eta\frac{\partial E_p}{\partial\theta_p} \quad (8)$$

where η is a positive constant called the learning constant. For the connection parameters σ, w and θ, we must differentiate with respect to the σ, w and θ which are deeply embedded in Eq.(6)(7)(8). Using the chain rule, we obtain

$$\frac{\partial E_p}{\partial\sigma_{p,ij}} = \frac{\partial E_p}{\partial O_p}\cdot\frac{\partial O_p}{\partial OR_p}\cdot\frac{\partial OR_p}{\partial AND_{p,j}}\cdot\frac{\partial AND_{p,j}}{\partial Y_{p,ij}}\cdot\frac{\partial Y_{p,ij}}{\partial\sigma_{p,ij}} \quad (9)$$

$$\frac{\partial E_p}{\partial w_p} = \frac{\partial E_p}{\partial O_p}\cdot\frac{\partial O_p}{\partial w_p} \quad (10)$$

$$\frac{\partial E_p}{\partial\theta_p} = \frac{\partial E_p}{\partial O_p}\cdot\frac{\partial O_p}{\partial\theta_p} \quad (11)$$

4. Simulation

We considered about learning-possibility that neuron model can recognize rotation in three-dimension. Thus, we simulated our neuron model to the directional selectivity problem in space. The state of input to the neuron model has two types(excitatory synapse, inhibitory synapse) of inputs by Koch's concept. The neuron model receives fast pathway inputs and slow pathway inputs with a delay time Δt(Figure 3). In the simulation, the learning parameters are set to $u = 3.0$, $v = 3.0$, $k = 20$, $\eta = 0.005$ and $w = 2.0$, $\theta = 0.5$. We used 4 AND gates. The neuron model before learning shows the randomly initialized morphology.The morphology is trained with the rotation movement. This movement is performed in the space, and the space is defined as (x,y,z)=(1~2,1~2,1~2). This movement is defined in the figure 4. In the figure 4, it is made to learn preferred direction by the training in the neuron model. The rotation movement stimuli are drawn as the component motion visible to the neuron model directly or with a delay Δt. This stimuli were used to train the neuron model by the error back-propagation algorithm in Eq.6-11. Figure 5 shows a success rate by the simulations, and ideal the value of error is very low. In the simulation, the neuron was trained to response only to preferred direction. Figure 6 shows

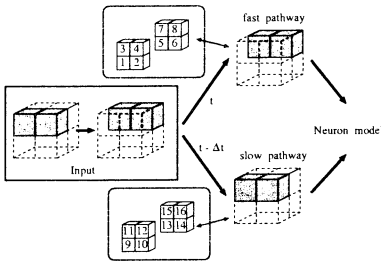


Figure 3: Inputs of neuron model.

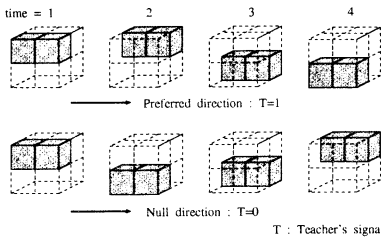


Figure 4: The rotation movement.

the state of output before learning, and after learning. As a result, the neuron model responds only to preferred direction, and it is impassive to null directions. Therefore, the neuron model is possible for learning the directional selectivity problem in space, and good performance was achieved. Figure 7 shows the morphology of the neuron model after learning. Neuron Layer was formed in w : 2.0 $\rightarrow$ 3.034, θ : 0.5 $\rightarrow$ 0.574 by learning. And, figure 8 shows outputs from Synapse Layer that was structured by the morphology of the neuron model. In the figure 8, the vertical axis shows synapse strength(over 0.5 is excitatory synapse, under 0.5 is inhibitory synapse), and the side shows the number of inputs(inputs:1~8, delay inputs:9~16). Excitatory synapse in delay input of Figure 8(e)(l)(o) and RMP in delayed input of (b) show the past pattern of the object. And, the other don't react by inhibiton of the shunting type. Therefore, we understand the direction of movement, and considers that the recognition of preferred direction is performed. We consider that the recognition of the rotation movement is performed by this morphology and three-dimensional movement can be recognized in MST area. It suggests that the simple neuron model may be possible to learn the solution to the directional selectivity problem and resemble real neurons quite well.

5. Conclusions

We have proposed a neuron model capable of learning for directional selectivity problem in space. The

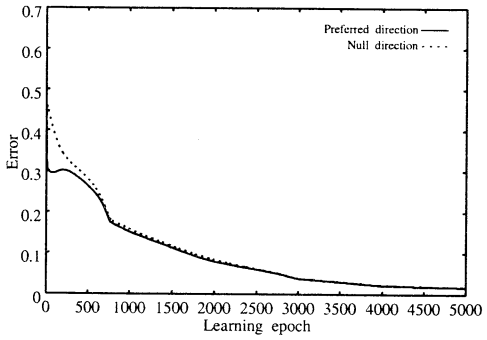


Figure 5: The error result of the training.

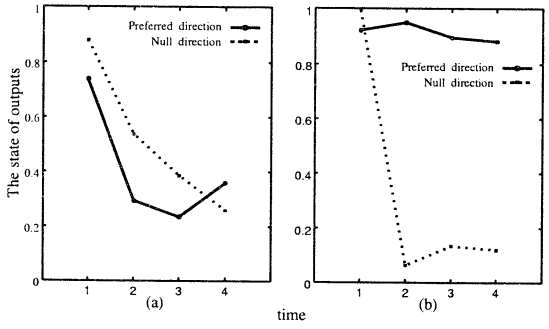


Figure 6: The output:(a)before learning, (b)after learning.

neuron model has interaction among synapses, and it receives two types inputs(normal and delay ones) by Koch's concept. We initialized the connections and the dendrites by random data, and trained them by the back-propagation algorithm for three-dimensional movement. As a result, the neuron system is successful in learning the directional selectivity. Furthermore, it is guessed the kind of synapse and the structure of dendrites by the morphology of the neuron after learning. The neuron model is considered to have the function which is near to the information processing of a real neuron. We guessed such a function exists in MST area. Therefore, it is considered that the neuron model is close to real neuron and give a good modeling for the MST area.

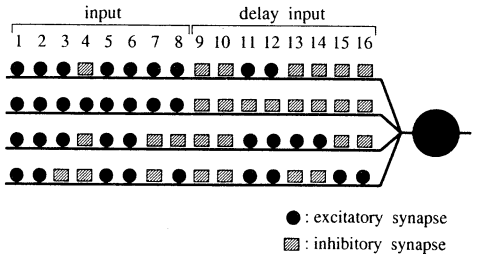


Figure 7: The morphology of the neuron after learning.

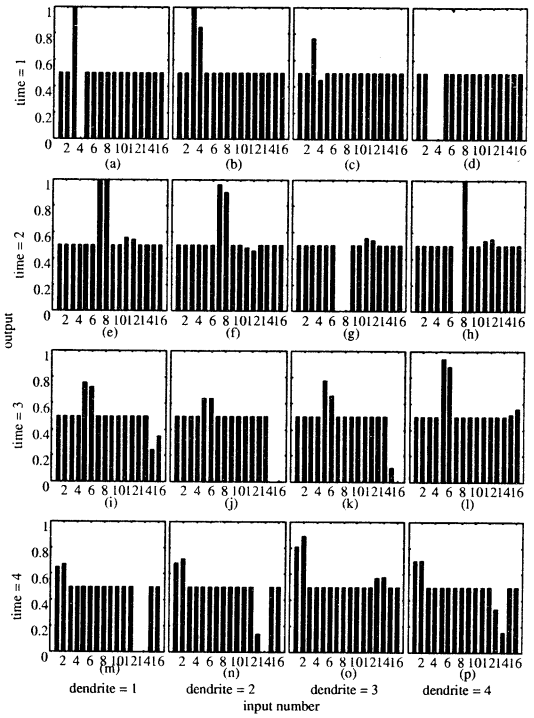


Figure 8: The output from Synapse Layer(Preferred direction).

References

[1] A. Mikami, " The evolution of visual perception, and brain ", Asakura Publishing Co. Ltd., Tokyo, 1993,3.

[2] Saito, H., Yukie, M., Tanaka, K., Hirosaka, K., Fukuda, Y. and Iwai, E., " Integration of Direction Signals of Image Motion in The Superior Temporal Sulcus of The Macaque Monkey ", J.Neurosci., vol.6, no.1, pp.145-157, 1986.

[3] G. Matsumoto, N. Ootsu, " Brain and computer 5 ", Baifu Publishing Co. Ltd., Tokyo, 1994,3.

[4] C. Koch, T. Poggio and V. Torre, " A Functional Interpretation Of Dendritic Morphology ", Phil. Trans. R. Soc. Lond. B 298,227-264

[5] Z. Tang, O. Ishizuka, M.Kuratsu and K. Tanno: "Binary learning machine using back-propagation Technique"–invited paper, Proc. *IEEE International Conference on Systems, Man and Cybernetics*, SanDiego, USA, TP7.6, Oct., 1998.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Autopoietic Active Selection by Cognitive Self-System : A unified theory for the origin and evolution of hierarchical neural-network-type cognitive biosystem-machines

*Koji Ohnishi and Hiroshi Shutou

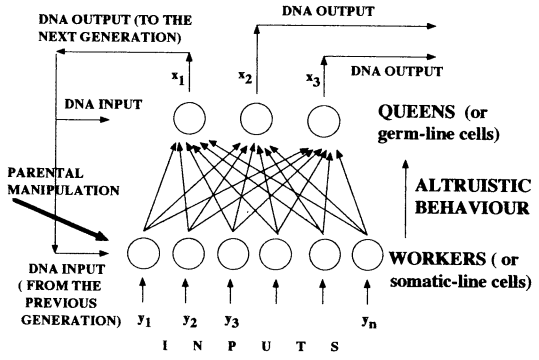
Department of Biology, Faculty of Science, Niigata University, Ikarashi-2, Niigata 950-2181, Japan . (E-mail: * ohnishi@sc.niigata-u.ac.jp)

Abstract: Well-made biomachines such as bee-super-organism, animal body, Mayr's diploid species, and the protein-making genetic machine are considered to have evolved as a self-learning neural network machine (SL-NNwM's) capable of functioning as a cognitive and/or thinking machine. Evolution of such biomachines ihas been achieved by active, autopoietic selection of randomly occurred mutations and variations suitable for improiving the NNwM. This cognitive selection is performed by the ability/ function of the cognitive NNwM. Accordingly, origin and evolution of life is a cognitive process of cognitively (= biotically ?) organized materials. Life is a kind of cognitive organization of matters.

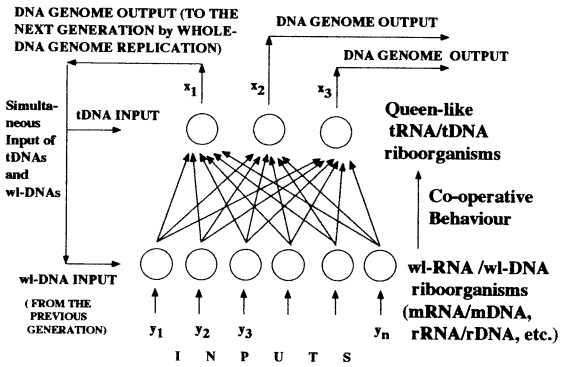
1. Introduction: Typical biomachines such as multi-cellular animal body, bee superorganism (SO) [both consistibg of queens (or germline-queen cells) and workers (somatic worker-cells)] and protein-making genetic machine (consisting of queen-like (ql-) tRNA ribo-organisms (RO's) and worker-like (wl-) mRNA/ rRNA RO's) have most plausibly emerged as SL-NNwM's, in which worker(-like) individuals and queen (-like) individuals are elements of an input-layer and an output-layer, respectively (See Models I and II in Fig. 1.) [1-3]. These SL-NNwM's are proposed to work as cognitive system in which informations from outer and inner environments are inputted and processed so as to efficiently output more of DNA base-sequence information to the next generation which then generate a feedback flow of DNA-information for making the next generation system. Thus every of these well-made biomachines (animal body, bee-eusociety, the protein-making /genetic machine) seem to work as a hierarchical NNwM possessing a function of cognitive system [2, 3].

In this paper, how and why hierarchical neural-network (NNw)-type cognitive systems are typical in biosystems, and how semeiotic maturation occurs in such systems. The final answer to this question is that such cognitive systems can autopoietically select mutations suitable for more adaptive, well-made systems.

2. Evolution of ribo-organisms : After early metabolic evolution in primitive soup, proto-cell would have evolved, in which 3-D-phosphoglycerates (3PG's) and therefrom derived biomolecules would have constituted early metabolic pathways, since 3-D-PG transporter protein possesses a most primitive amino acid (aa) sequence closely homologous to the "*trrnD*-peptide" [1-5]. The ATP/ADP/AMP-type energy-acquiring system would have been therafter established, and earliest small RNA replicators (= intracellular RO) would have emerged (by olygomerization of AMP). These RNA replicators are newly emerged organisms (= RO's) living in intracellular environment. Some kind of RO (= RNA) might have evolved to be capable of being aminoacylated (at its 5'-CCA end), and have possessed a ribozymic activity for catalyzing di- and/or tri-peptide formation. Such presumptive tRNAs could have further evolved to be an RNA (proto-tRNA) commonly ancestral to all of the contemporary tRNAs. Diversification of proto-tRNA ROs must have occurred , which generated different tRNA species, each of which can interact with its respectively specific aa. At sometime befor the emergence of genetic codes, RO's had acquired a life-cycle consisting of an RNA-phase and a DNA-phase, the latter being a replicatice phase. An RNA replicator or a RO is a cognitive, living entity, and contemporary tRNA ROs recognize codons on mRNA. Anticodon and specific aa-binding capability of tRNA are the signifier (*signifian*) and the signified (*signifie'*) in de Sausseur terminology [2, 3]. The correspondence between anticodon (*signifian*) and the aa-specificity (*signifie'*) is mostly arbitrary in contemporary tRNAs. However, at the biginning of aa-specificity, the specificity must have been caused by some stereochemical, key-keyhole rerationships. The arbitrariness in the anticodon-aaspecificity correspondences could have maturated by making aa-specific protein-machine (= aminoacyl-tRNA synthetase (ARS)) made by tRNA RO's using triplet codon system, which is a most thouroughly maturated semeiotic system in organic world. This means that genetic codon system is a most typical semeiotic cultural system which is a culture of the hierarchical RO'ic society. The arbitrarines in signifian-signifie correspondence is the result of semeiotic maturation commonly found in general semeiosis such as human language system.



[Model I] Hierarchical Altruistic Behavioral Network of Queen and Worker Kin Individuals (Bee Super-Organism and Multicellular Animal Body)



[Model II] Hierarchical Co-operative Behavioral Network of Queen-like tRNAs and worker-like (wl-) RNA riboorganisms (Intracellular Riboorganismic Society)

Fig. 1. DNA-information flow in the hierarchical behavioral network in altruistic kin societies (Model I) and in intra-cellular co-operative tRNA/tDNA ribo-organismic society (Model II). Based on [1,2].

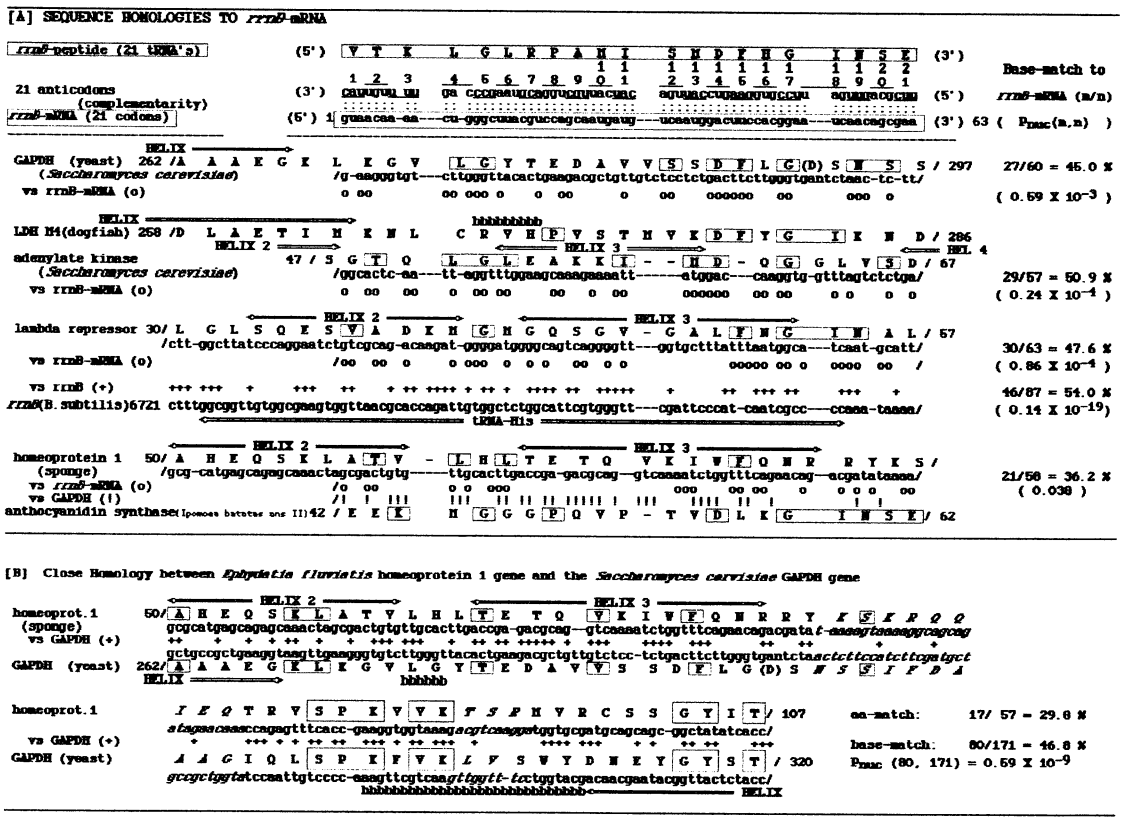


Fig. 2. Alignment of protein sequences related to a hypothetical 21-aa *rrnB*-peptide. Aa's identical to the rrnB-peptide are boxed. GAPDH = glyceraldehyde 3-phosphate dehydrogenase, LDH = lactate dehydrogenase. "bbbbb" indicates beta-sheet. Base identities are indicated by "o", "+", and "!". , Data are from GenBank and/or EMBL Databases. Pnuc(m,n) denotes a probability that base-identities occur at m out of the n-aligned base-positions by chance, assuming equal occurences of 4 different bases.

3. Further evidence for Poly-tRNA theory: The most difficult, basic portion of the problem of the origin of mRNAs and genetic codes seems to have been solved by the analyses based on the poly-tRNA hypothesis [1-6]. In this section, further evidence is presented for proving the poly-tRNA hypothesis.

B. subtilis (BS) possesses two types of poly-tRNA structure, one being a *trrnD*-type 16-tRNA structure found in the BS *trrnD* operon and in other gram-positive bacterial operons. The other is an *rrnB*-type poly-tRNA found in the BS *rrnB*-operon The *rrnB* operon-type poly-tRNA consists of 21 tRNAs whose aa specificities are in the order of "(5')VTKLGLRPAMISMDFHGIN SE (3')" (Fig.3). This poly-tRNA structure has been proposed to be a relic of an early protein-synthesizing poly-tRNA molecule [4,6]. As a result of homology-search and Harr-plot matrix analyses, a hypothetical 21-aa-peptide having this aa-ordering (*rrnB*-peptide, See Fig. 2) were found to be homologous to glycelaldehyde-3-phosphate dehydrogenase (GAPDH) and the helix-turn-helix DNA-binding domains in lambda-repressor and homeoproteins. A hypothetical 63-base-RNA (*rrnB*-mRNA) comprizing 21 triplets, each being complementary to the corresponding anticodon in the *rrnB*-poly-tRNA. The *rrnB*-mRNA could have been an earliest mRNA encoding a *rrnB*-peptide-like protein. Harr-plot analysis revealed that the $tRNA^{His}$ region of the *rrnB*-poly-tRNA is most plausibly homologous to the *rrnB*-mRNA and the corresponding region of the cI gene encoding lambda repressor. The resulting alignment is shown in Fig. 2, clearly confirming that an earliest mRNA would have evolved from some tRNA ($tRNA^{His}$) by natural selection of those base-replacement-mutations capable of generating Wason-Crick-type base-pairings between $tRNA^{His}$ and anticodons. It is very interesting to note that the *rrnB*-peptide homologue regions possess α-helix structure in most cases as found in the helix-3 of adenylate kinase, hekix-3 of lambda repressor and homeoproteins (See Fig. 2). For evaluating homology levels, P_{nuc} (m,n) values [5,6,1] were computed and given in Fig. 2.

4. BPM Computer-Simulation of a learning-NNwM model of qw-type hierarchical society :

Bee-eusociety and animal body are both hierarchical society of qw-type. As has been already discussed [2,3], the DNA-information flow shown in the Mode l of Fig. 1 satisfies the conditions of two-layered NNwM whose input layer and output layer are workers (or somatic cells) and queens (= worker-bee's yonger sisters) (or germ-line cells), respectively (See details in refs. [2, 3]). Thus the Model I -type hierarchical society-machine would have functioned as a learning-NNwM. Since queens/ germ-line cells are organic individuals closely kin to workers (genetic relatedness = 3/4 in bee-society, and = 1 in animal body), altruistic behaviour [7] makes a DNA-information-flow from workers to queens (= worker-bee's younger sisters, or germ-line cells) , and from qeens to the next generstion system. Parental manipulation (Fig. 1) such as the (behavioral and/or chemical) manipulation by the previous-generation queen (in bees and wasps) or previous generation mRNA (*bicoid* mRNA, etc. in fruit-fly) would bring out the first making of the sterile worker-individuals. Thus, structural similarity between hymenopteran SO (= eusociety) and animal body (SO of unicell-anomals) is not a superfacial one, but a more essential, fundamental one, suggesting that both bio-machines could have evolved by a common evolutionary logic.

In order to analyze learning process of the neural net-work (NNw)-model I in Fig. 1, a simple computer simulation was made by back propagation method (BPM) [8, 9]. A most simple two-layered hierarchical learning-NNw consisting of one queen (an output-layer unit) and two workers (input-layer units) (Fig.3, A) was considered. The input informations (at time t) to the input layer units are randomly selected values of $y_1^{(t)}$ and $y_2^{(t)}$ ($0 \leq y_i^{(t)} \leq 1$), and the input to the queen is given by $z(t) = c_1(t)\, y_1^{(t)} + c_2^{(t)} y_2^{(t)}$, where $c_i^{(t)}$ denotes conection weight (at time t) between the queen and the i-th worker. The output from the queen is given by $x^{(t)} = h(z^{(t)}) = 1 / \{ 1 + \exp[-\beta(z^{(t)} - \theta)] \}$, where h(.) denotes a sigmoid function (x = 0.5 for z = θ). Category p input-information (p = 1, 2) were defined by $z_{p0} = c_{10} y_{1p} + c_{20} y_{2p}$ (c_{10} = 1.25, c_{20} = 2.0), satisfying $z_1 \geq 1$ and $z_2 < 1$, respectively. Teacher signal for category 1 is given by $d_1 = h(z_{10})$, and error function is given by $E(\mathbf{c}) = (1/2)(x_1 - d_1)^2$. Connection weight at (t + 1)-th time is modified by

$$c_i(t+1) = c_i(t) - (1-\alpha)\eta \left[\partial E / \partial c_i\right]_t + \alpha(c_i(t) - c_i(t-1)) \qquad \text{[Eq. 1]}$$

where $[\partial E / \partial c_i]_t$ denotes $\partial E / \partial c_i$ at $c_i = c_i(t)$, and $\alpha(c_i(t) - c_i(t-1))$ is an item of inertia. Repetition of computation by [Eq. 1] was continued up to $t = t_{end}$, where $E(\mathbf{c}(t)) < 10^{-6}$. Simulation for 40 trials by [Eq. 1], employing α = 0.2, β = 2.0, and η = 3.0, resulted in [mean ± S.D. of t_{end}] = 23.5 ± 13.1 (Fig.3, B). Corresponding error function of $\mathbf{c} = (c_1, c_2)$, defined by $E_{\mathbf{c}}^{(t)} = (c_1^{(t)} - c_{10})^2 + (c_2^{(t)} - c_{20})^2$, was also found to have rapidly reduced to be $< 10^{-6}$ (Fig.3, C), meaning that learning for descriminating category 1 and 2 was very rapidly achieved as early as t_{end} = 23.5. Thus it is concluded that even a simplest two-layerd NNW (Fig.3, A) can work as a L-NNwM capable of achieving an efficient descrimination of linearly separable data-sets. In real case of the NNw-model I (in Fig. 1), every element in both layers area biotic individual which itself is a

cognitive SL-NNwM. This means that every element of the simplest qw-type machine in Fig. 3[A] can be redrawed as shown in Fig. 4, in which every element in machinogenesis throughout evolution in NNw-models I (or II) is a kind of generalized "thinking" process. Human thinking by brain (which is often accompanied by

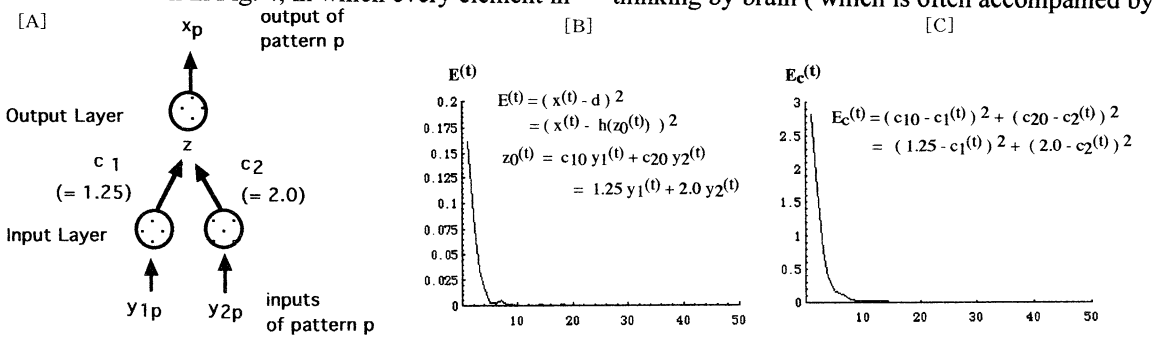


Fig. 3. B back propagation simulation of the two-layered NNW ([A]) consisting of one queen (output- layer unit) and two workers (input-layer units). See text for further details.

Fig. 3A has its own intra-individual L-NNw consisting of (at least) two layers. Thus the model in Fig.4 consists of 4 layers including two "hidden layeres", by which the NNwM can descriminate non-linearly separable data-sets. This would sexplain the reason why early two-layerd (qw-type) sociogenesis (as in wasp society) could have rapidly acquired the feature of L-NNwM. Since the teacher signal, d_1, is considered to be possessed by the biotic individuals (queens and/or workers), the L-NNW would most plausibly function as a self-larning NNwM (SL-NNwM).

Thus we can conclude that the qw-type NNwM was made by self-improving and self-making based on the function of the SL- NNw. Accordingly, even if mutations occur randomly, mutated genes which are more adaptive would have been actively selected by the SL-NNwM. Essential basic similarities between model I and model II in Fig. 1 suggest that genetic apparatus would have been made by essentially similar evolutionary logic. The SL-NNw's in I and II in Fig. 1 strongly suggest that machine-making by self-learning and self-

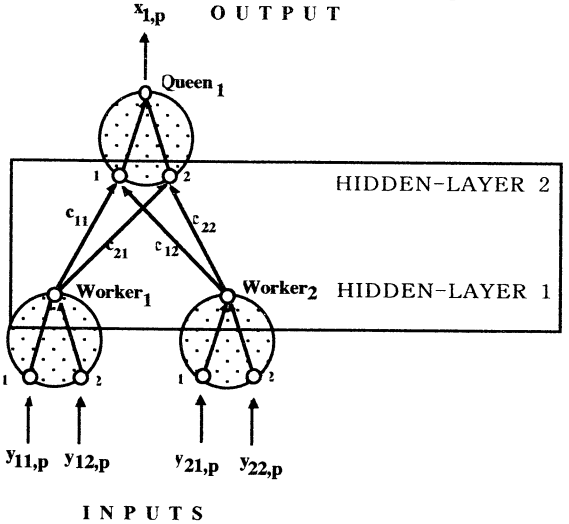


Fig. 4. Double-two-layered NNwM.

"conciousness") is a special case of these generalized thinking phenomena. These results well coinside with the previously proposed concept of "generalized thinking" and "generalized culture" in various hierarchical levels of biosocieties.[12].

5. Bee-dance language system as a nerve-like cognitive system of the bee-superorganism : In the well-known *Aplasia* simple nervous system, a sensory neuron (connecting siphon and inputting water-flow information) makes a direct synapsis onto the moter-neuron (connecting gill-muscle and outputting mustle-contracting information), with a feedback connection by an interneuron [8]. This sign system is capable of making habituation and sensitization. This can be considered to represent an earliest evolutionary step of the "brain" or brain-like "thinking sistem". If we compare this system with the bee-dance language system, striking similarities between them can be ruled out, as shown in Fig. 5. In the bee-SO, some worker-bees (sensory bee) find food (flower), and the information concerning the site of food (direction and distance from hive) is trnamitted to other worker-bees (motor-bees) by doing the so-called "bee-dance", and the motor-bees recognize the information, and then fly to the food-site. Sensory-bees and motor bees are worker individuals of the qw-type eusociety (= bee-SO) , which are comparable to the sensory neuron and the motor-neuron in the *Aplasia* system, which are worker-type unicell animals of the qw-sosiety (= SO of unicell-animals). Thus the dance-language is a cognitive semeiotic system comparable to some kind of semeiotic phenomenon performed in the synaptic transmission in *Aplasis* simple nervous sistem, and also to some synaptic signe-systems (chemical and/or electric signes) in more highly evolved manmalian brains. Animal neuronal synaptic system would have evolved by natural selection of multicellular animal individual (= SO of unicell animals) based on Darwinian fitness of multicellular individual. Accordingly, it is almost undoubtedly concluded that the semeiotic maturation of the bee-dance system have evolved by

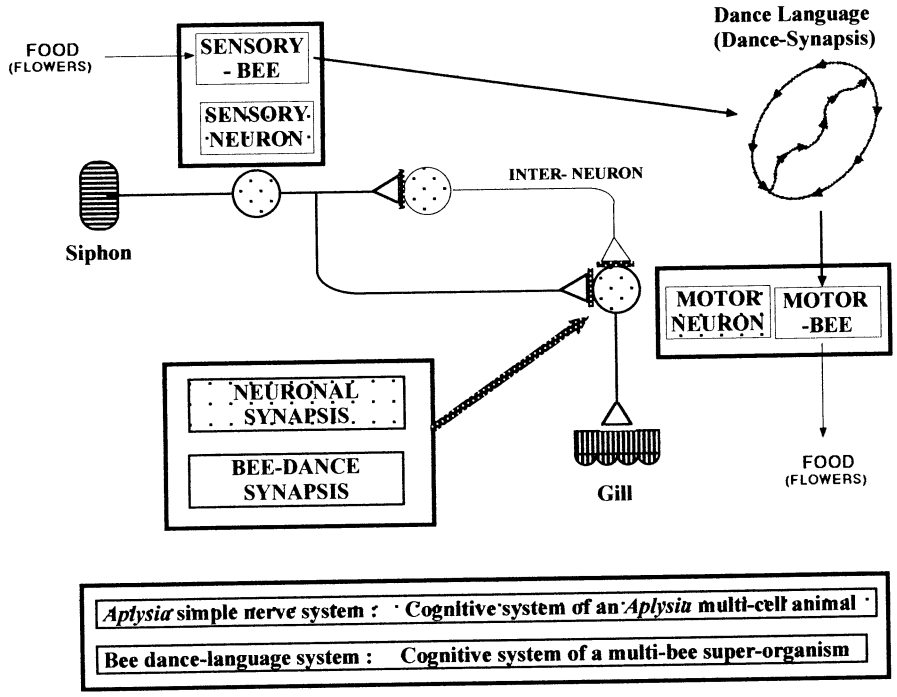


Fig. 5. Close similarities between the *Aplasia* simple neuronal system and the bee-dance system.

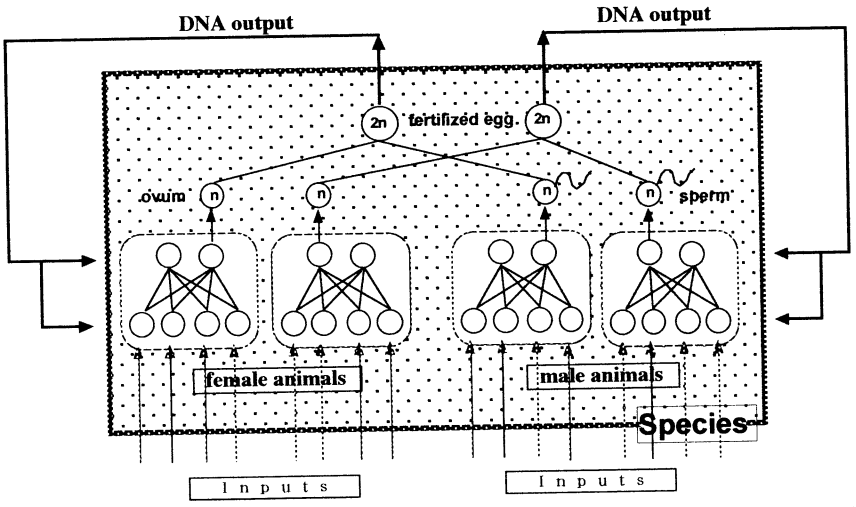


Fig. 6. A schematic model of Mayr's diploid species as a probablistic neural network machine. Males and females are multicellular (animal-)bodies which are also learning-neural network machines.

selection pressure of Darwinian fitness of bee-SO, and not by that of multicellular bee-individuals. This conclution would elucidate how semeiotic maturation could have been selected coupled with acceration of evolutionary bio-machinogenesis of hierarchical societies.

During war-periods of human societies, various kinds of semeiotic or sign systems tend to arise and have more important roles in making a hierarchical military system, than in ordinary society-systems at peace-periods. Such military sysytem is a kind of transient sociomachinogenesis, and woulf give highr probability of the survival of the society, which would be related to the above-mentioned semeiotic maturation in hierarchi-cal sociogenesis and machinogenesis. In every of these machinogenetic process of hierarchical societies, communications between constituting individuals become replaced by

sign-systems which ensures smooth tramnsmission of information. The constituting individuals tend to lost free-living capabilities and become elements of the society-machine, which would be a most important and essential feature in machinogenesis and the survival of the machine-like society. The unit of natural selection is "hierarchical society" in these cases.

6. Origin(s) of primate language systems: A hypothesis has been proposed that the velvet monkey society possessing three different alarm voice-signs (each corresponding to "snake", "eagle", and "leopard", respectively) is the origin of primate language system. In this case, the society is a SO-like quesi-(or transient) individual, and the alarm-making monkey is a sensory organ (or "sensory monkey" comparable to "sensory bee") of the society, whose alarm-sign is efficiently transmitted to other society-members (= "motor monkeys" in analogy to bee-eusociety). The semeiotic maturation would have been achieved by selection pressure to the society-quesi-individual, rather than to the monkey individuals. The basic rule in the emergence and maturation of alarm sign-system is considerably similar to the bee-dance sign system discussed above.

7. Existing entity of Mayr's diploid species as a NNwM : Mayr [10] defined diploid species as "groups of actually or potentially interbreeding natural populations which are reploductively isolated from other such group". This definition well coincides with a hypothesis that diploid species is an entity as a probablistic SL-NNwM shown in Fig. 6, in which females and males are input-layer elements, and gametes (fertilized eggs) are output-layer elements. Each gamete is generated by combining an input egg and an input sperm outputted from a female and a male, respectively, and reconstitute the next generation by means of a feed-back information -flow from gametes to the input-layer elements of the next generation system. Thus "species" is also a hierarchical SL-NNwM which recognizes informations and/ or stimuli from surrounding and intra-specific environments, and outputs gametes to the next generation. Accordingly, "species" is a unit of natural selection whose Darwinian fitness depends on connection weits of the probablistic NNwM. Semeiotic signals in mating behaviours would have generated better connection-weights for improving the NNwM(="species-machine"). This means that "selection" is not a mere passive process, but considerably depends on individual's active behaviours, suggesting "active, self-improving evolution" or "autopoietic evolution" of dioploid organisms. Diploid cell is also considered to have emerged by a kiship-recognizing cognitive behaviours (= homologous chromo-some-pairing) of two haploid unicell-organisms [11,12].

8. A unified theory of autopoietic selection by cognitive self-system : As discused above, various biosystems seem to have evolved as SL-NNwM's in which variations or mutations randomly occured in the previous generation tend to be selected by the biosystem (= self-machine) of the next generation. As typically found in the case of Model I in Fig. 1, random mutations are generated in the queens/germ-line-cells of the prrevious generation NNwM, and the mutations are autopoietically selected by the workers/somatic-cells in the present generation NNwM. Thus germ-line cells are mutation-generators, and somatic cells are mutation-selectors. This generator-selector machine constitutes an autopoietic, sef-improving SL-NNwM such as an animal body. This conclution is considerably differs from Dawkin's conclution [13] that well-made biomachines are results of random mutations and random (natural) selections.

References

[1] Ohnishi, K. , Hokari, S., Yanagawa, H.(1999), Origin and evolution of early-peptide synthesizing biomachines. In: Sugisaka, M.(ed.), Proc. of the 4th Int. Symp. on Artificial Life and Robotics, 344-349.

[2] Ohnishi, K. , Kanbe, D.(2000), Neural network-like hierarchical sociogenesis as a common logic underlying evolutionary semeiogenesis and machinogenesis in various levels of organic individuals. In: Sugisaka, M.(ed.), Proc. of the 5th Int. Symp. on Artificial Life and Robotics, 573-575.

[3] Ohnishi, K.(2000), Hokari, S., Kanbe, D.: Genesis of genetic codes and other semeiotic systems by means of neural-network-like hierarchical sociogenesis and bio-machinogenesis. In: Proc. of the First Int. Conference on Systems Biology (Organized by Japan Science and Technology Corporation) , 203-209

[4] Ohnishi, K (1993), Evolution from semi-tRNA to early protein-synthesizing RNA molecule. In: Satoh, S. et al. (ed.), *Endocytobiology V* , 407-414,Tubingen Univ. Press, Tubingen.

[5] Ohnishi, K (1993): Poly-tRNA theory on the origin and evolution of mRNA and genetic codes. In: Takagi,T et al.(eds.), *Genome Informatics IV*, 325-331, University Academy Pr.,Tokyo.

[6] Ohnishi, K., Tanaka, H., and Yanagawa, H. (1998): The origin of DNA-binding domains, as viewed from poly-tRNA theory *Nucleic Acids Res., Symp. Ser. (Oxford),* 39: 251-252.

[7] Hamilton, WD.(1964): The evolution of social behaviour. J. Theoret. Biol. 7: 1-52.

[8] Rowe,GW. (1.997), *Theoretical Models in Biology* , Clarendon Press, Oxford.

[9] Dracopoulos, DC (1997), Evolutionary Learning Algorithms for Neural Adaptive Control. Springer, London .

[10] Mayr, E (1940).: Speciation phenomenon in birds. Amer.Naturalist 74: 249-278.

[11] Ohnishi, K.(1996), On the origin of diploidy and diploid species: Homologous chromosome-pairing as kinship-recognizing co-operative behaviour of unicellular haploid individuals. *Origins of Life*, 26: 485-486.

[12] Ohnishi, K.(1990), Evolutionary meanings of the primary and secondary structure of the "ur-RNA". In: *Symmetries in Science IV* (ed. by B. Gruber and J.H.Yopp), 147-176, Plenum, New York.

[13] Dawkins,R (1986), *Blind Watchmaker.* Longman , London.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Improved adaptive fuzzy control and its application in molten carbonate fuel cell

o Xing-jin Sun, Guang-yi Cao, Xin-jian Zhu
Institute of Fuel Cell
Shanghai Jiaotong University
Shanghai 200030, P.R.China

Abstract

Having improved a back-propagation algorithm and a nearest neighbor clustering algorithm, we apply them respectively to build two different adaptive regulators: a parameter regulating module and a cluster analyzing module, which organically combined with a controlling module compose an improved adaptive fuzzy control system. A real-time adaptive regulation of both the parameters and the system structure is accomplished with the available language information as well as the running data. Such an adaptive fuzzy controller will be applicable to nonlinear systems with distributed parameters.

Molten carbonate fuel cell (MCFC) stack is a nonlinear distributed parameter system, whose analytical solution is not available. So traditional control technology is incapable of obtaining the optimal result for the MCFC stack. While the adaptive fuzzy control technology is able to utilize the real-time data as well as the laws educed from the numerical result. Finally, we compare the proposed improved adaptive fuzzy controller with the traditional PID controller as well as the traditional fuzzy controller respectively. Simulation results show that the proposed controller is better than the other two ones in the overtune, the stabilization rate, and the stable precision.

Keywords

ADAPTIVE FUZZY CONTROL - BACK PROPAGATION ALGORITHM - NEAREST NEIGHBOR CLUSTERING ALGORITHM - MOLTEN CARBONATE FUEL CELL (MCFC)

1 Introduction

The adaptive fuzzy logic system is a fuzzy system with capability of learning. An adaptive fuzzy controller designed according to the adaptive fuzzy logic system brings an effective control method[1].

Aiming at defaults of Mamdani method, people have added other technology such as neural network and genetic algorithm to fuzzy system[2,3]. One of the key algorithms is BP algorithm, which is applicable to any forward network. So, if we express the fuzzy logic system as a forward network system, then we can apply the BP algorithm to regulate parameters of the fuzzy logic system. On the other hand, neighbor clustering algorithm is capable of learning sample data on-line, which brings us a way of regulating fuzzy rule database. To combine regulation of fuzzy control rules' parameters and on-line amending of rule database, Xu[3] proposed a weighted summing scheme, which, however, divided the system into two independent subsystems. We will try another scheme to utilize the potential of the algorithms.

Development of fuzzy logic system theory supplies powerful tool for fuzzy controller structural analysis, with which could we apply various mature theory and technology to improve the traditional fuzzy control system as well as solve difficulties such as stability, astringency and so on in the evolution of fuzzy control[4].

Molten carbonate fuel cell (MCFC) transforms chemical energy within the fuel gas and the oxidant gas into electricity continuously without combustion. It is regarded as one of the most promising power generating systems in the future[5]. As the external load changes, the current and then the output voltage of the MCFC will also change, which leads to an undulation of MCFC temperature[6]. Commercialization of the MCFC calls for an excellent performance, a high security and a long lifetime. However, the MCFC stack is a nonlinear distributed parameter system, whose analytical solution is not available[7]. So traditional control technology is incapable of obtaining the optimal result for the MCFC stack. While the adaptive fuzzy control technology is able to utilize the real-time data as well as the laws deduced from the numerical result. So the adaptive fuzzy control is capable of controlling of MCFC stack.

2 The adaptive fuzzy control system of MCFC stack

According to mathematical solution and simulation of MCFC stack mathematical model, we can conclude that as the external load changes, the current follows it, which in turn leads to the change of output voltage and current density distribution, and finally the undulation of temperature. We will adopted the fuel gas flux, u as the control variable to maintain the output voltage, y, namely, deduce the difference, e, between y and control goal $\overline{y}$. At the same time, we should minimize temperature undulation and mend temperature dynamic characteristic to the most.

First, based on the analysis results and practical operation experience, we deduced twenty initial fuzzy rules to establish a proper fuzzy rule database. Adopting single value fuzzifying, center average de-fuzzifying, product reasoning, and Gause membership function, we established the initial fuzzy logic system. During the procedure of control, on the one hand, the improved BP algorithm amends parameters of the fuzzy logic system real time, on the other hand, the nearest neighbor clustering algorithm utilizes the running data to renew the fuzzy rule database.

3 The adaptive fuzzy controller of MCFC stack

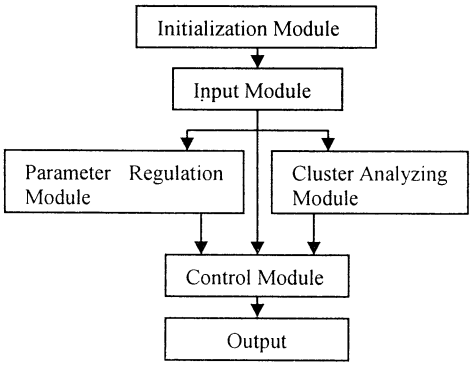


Fig.1. Adaptive Fuzzy Controller diagram

The design scheme of the whole adaptive fuzzy controller is shown in Fig.1. It is made of initialization module, input module, parameter regulation module, cluster analyzing module and control module.

3.1 Initialization module and input module

According the initial rules, the initialization module defines initial parameters $\bar{u}^l$, $\bar{e}^l$, $\Delta\bar{e}^l$, σ^l and σ^l_Δ, initial conditions e_0, e_{-1} and u_0, control goal $\bar{y}$, step Δt, cluster radius r and σ. The input module reads e_n、e_{n-1} and u_n.

3.2 parameter regulation module adopting BP algorithm

The fuzzy logic system is shown as Eq. (1). The aim is to regulate parameters $\bar{u}^l, \bar{x}^l_i$ and σ^l_i so as to minimize Eq.(2).

$$f(\boldsymbol{x}) = \frac{\sum_{l=1}^{M} \bar{y}^l \left(\prod_{i=1}^{n} a_i^l \exp\left[-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l} \right)^2 \right] \right)}{\sum_{l=1}^{M} \left(\prod_{i=1}^{n} a_i^l \exp\left[-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l} \right)^2 \right] \right)} \quad (1)$$

$$J = \frac{1}{2}(y_n - \bar{y})^2 \quad (2)$$

Let $(\boldsymbol{x}_n, u_n)$ be a pair of known input-output data, where $\boldsymbol{x}_n \in U \subset R^k$, $u_n \in V \subset R$, $a_i^l = 1$. Let M be the number of initial rules, which is 20 here. We regulate $\bar{u}^l$ by Eq.(3)

$$\bar{u}^l(n+1) = \bar{u}^l(n) - \alpha \frac{\partial J}{\partial \bar{u}^l}\Big|_n$$

, $l = 1,2,\ldots, M$; $n = 0,1,2,\ldots$ (3)

where α denotes step. Let

$$z^l = \prod_{i=1}^{n} \exp\left[-\left(\frac{x_i - \bar{x}_i^l}{\sigma_i^l} \right)^2 \right],$$

$$a = \sum_{l=1}^{M} (\bar{u}^l z^l), b = \sum_{l=1}^{M} z^l \quad (4)$$

So we get relation $f = a / b$. Here J only depends on $\bar{u}^l$ via f in the expression of a. By the chain method, we get

$$\frac{\partial J}{\partial \bar{u}^l} = e_n(e_n - e_{n-1}) \frac{\partial f}{\partial a} \frac{\partial a}{\partial \bar{u}^l} = e_n(e_n - e_{n-1}) \frac{z^l}{b} \quad (5)$$

According to Eq.(5) and (3), we deduce learning form of $\bar{u}^l$

$$\bar{u}^l(n+1) = \bar{u}^l(n) - \alpha \frac{e_n(e_n - e_{n-1})}{b} z^l$$

, $l = 1,2,\ldots, M$; $n = 0,1,2,\ldots$ (6)

Similarly, $\bar{x}^l_i$ is regulated by

$$\bar{x}_i^l(n+1) = \bar{x}_i^l(n) - \alpha \frac{\partial J}{\partial \bar{x}_i^l}\Big|_n$$

, $i = 1,2,\ldots,k$; $l = 1,2,\ldots, M$; $n = 0,1,2,\ldots$ (7)

Here J only depends on $\bar{x}^l_i$ via f in the expression of z^l. By the chain method, we get learning form of $\bar{x}^l_i$

$$\bar{x}_i^l(n+1) = \bar{x}_i^l(n) - e_n \alpha (e_n - e_{n-1}) \frac{\bar{u}^l - f}{b} z^l \frac{2\left(x_i(n) - \bar{x}_i^l(n)\right)}{\left(\sigma_i^l(n)\right)^2}$$

, $i = 1,2,\ldots,k$; $l = 1,2,\ldots, M$; $n = 0,1,2,\ldots$ (8)

σ^l_i is regulated as

$$\sigma_i^l(n+1) = \sigma_i^l(n) - \alpha \frac{\partial J}{\partial \sigma_i^l}\Big|_n$$

, $i = 1,2,\ldots,k$; $l = 1,2,\ldots, M$; $n = 0,1,2,\ldots$ (9)

Here J only depends on σ^l_i via f in the expression of z^l. By the chain method, we get learning form of σ^l_i

$$\sigma_i^l(n+1) = \sigma_i^l(n) - e_n(e_n - e_{n-1}) \frac{\bar{u}^l - f}{b} z^l \frac{2\left(x_i(n) - \bar{x}_i^l(n)\right)^2}{\left(\sigma_i^l(n)\right)^3}$$

, $i = 1,2,\ldots,k$; $l = 1,2,\ldots, M$; $n = 0,1,2,\ldots$ (10)

Actually, Eqs. (6), (8) and (10) represent the BP algorithm, where, because of the holder, parameters are regulated based on the input-output data of the previous step, and the normal error is back propagated to related parameter process unit with a step of delay. Since the initial fuzzy logic system is set up, we needn't calculate parameters in the beginning. Once the control is performed, the parameters of the whole fuzzy system, including the subsystem defined by the initial rules and the one defined by neighbor clustering algorithm, are regulated real time.

3.3 Cluster analyzing module adopting the nearest neighbor clustering algorithm

As the external load changes, the MCFC stack model will changes also. Once the error is beyond the knowledge of initial rules, and system performance deteriorates only with the regulation of parameters of the initial subsystem, the cluster analyzing module should be added to adjust the rule database real time so as to strengthen the regulation of the system.

The nearest neighbor clustering algorithm decides whether to set up a new cluster, ie., rule, according to the distance between the data and the center of clusters. While the cluster analyzing module groups the sample data with the nearest clustering algorithm, treats each pair of data as a cluster, decide whether to refresh the rule database, and regulate parameters with the BP algorithm at last.

Considering data pair (x_n, u_n), n= 0,1,2,..., we assume that the number of clusters is M, whose centers are $x_{n+1}^1, x_{n+1}^2, \ldots, x_{n+1}^M$ rspectively. Calculate $\left|x_n - x_{n+1}^l\right|$ (l = 1,2,..., M), and x_n is named the nearest cluster of a certain x_{n+1}^l if $\left|x_n - x_{n+1}^l\right|$ is the minimum one.

If $\left|x_n - x_{n+1}^l\right| \le r$, no new clusters should be added, or else a new one should be added. Resuming that it is the first time to add a cluster, we add a new cluster center at x_n, and make $\bar{x}_{n+1}^{M+1} = x_n$, $\bar{u}_{n+1}^{M+1} = u_n$, $\sigma_{i,n+1}^{M+1} = \sigma$. Since then, each time a new cluster is added, the nearest neighbor cluster of x_n should be defined, namely, double cluster analysis should be performed.

The value of radius r determines the complexity of the system. The smaller r is, the more clusters we have, which means a heavier calculation. To prevent too complex a system, we set an oblivion gene to limit the number of clusters. We resume that the initial rules are scientific, and the number of clusters generated by them should be kept constant. Hence only the number of newly generated clusters, N, should be limited. Here we make N=40. In addition, to make the new rules exactly describe the system the whole way, not only should we add or discard some rules, but reserve those describing common characteristic.

Each time after a cluster analysis is performed, we move backward the clusters related to the new rules so as to refresh the rule base. For a newly generated cluster, if it is not demonstrated to be a nearest cluster of a group of data during a oblivion cycle, we will discard it. In this period, if there is a cluster becomes to be the nearest cluster of a new sample data, it will be generated again.

3.4 Design of Control module

The parameter regulating module and the cluster analyzing module works dependently. The BP algorithm simultaneously processes both cluster parameters of initial rules and those of the cluster got at the previous step by the cluster analyzing module, and then delivers the results to cluster analyzing module and the control module respectively. While each clustering process is based on the parameters offered by the parameter regulating module and the input-output sample data.

With the refreshed parameters of the regulated fuzzy system by the parameter regulating module and the cluster analyzing module, we can deduce z^l, a and b according to Eq.(4), and control variable $u = f(x) = a/b$. The scheme of the control module is shown in Fig.2.

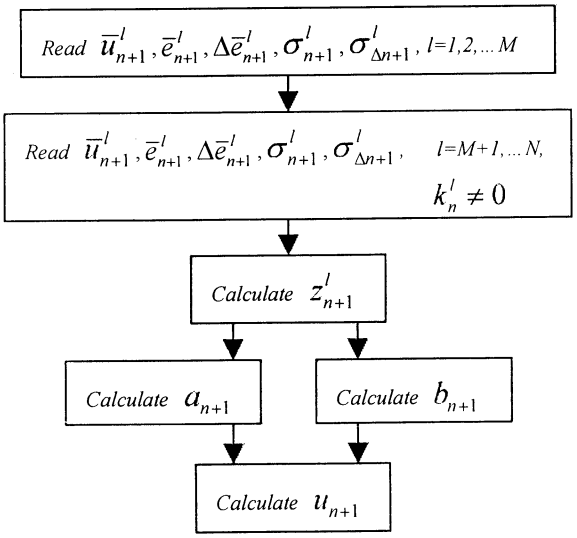


Fig.2. Scheme of the control module

4 Simulation and discussion

The above adaptive fuzzy controller is applied to the MCFC stack so as to maintain the output voltage in the limitation of its temperature by the control of fuel gas flux. Assume that the stack load has a step change, the goal of the output voltage is 8 Volt, initial fuel gas flux is 30m/s, and the step is 0.1s.

First, we compare the results of improved adaptive fuzzy control with those of PID control, which is shown in Fig.3. We can conclude that the former has better performance on the output voltage than the latter.

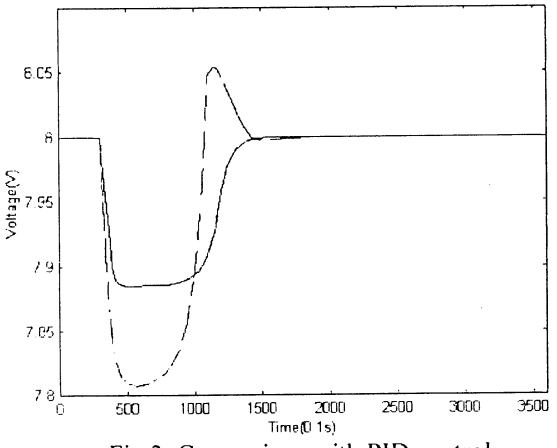


Fig.3. Comparison with PID control

---- PID control —— Improved adaptive fuzzy control

Fig.4 shows the comparison of the improved adaptive fuzzy control with the traditional fuzzy control with Mamdani table look-up technology.

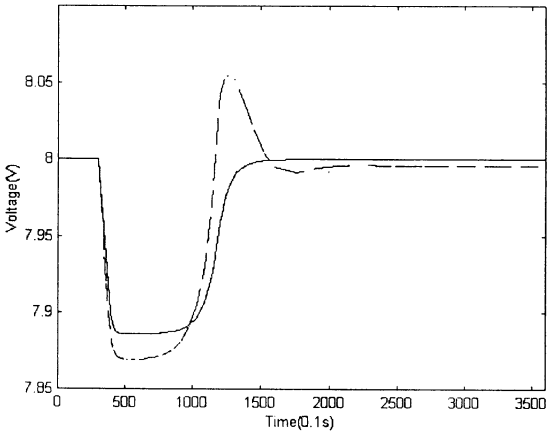


Fig.4. Comparison with traditional fuzzy control
---- Traditional fuzzy control
—— Improved adaptive fuzzy control

5 Conclusion

We apply an improved back-propagation algorithm and a nearest neighbor clustering algorithm respectively to build a parameter regulating module and a cluster analyzing module, which organically combined with a controlling module compose an improved adaptive fuzzy control system. Based on rules deduced from the analysis of performance of the plant, an initial fuzzy logic system is built, whose parameters then needn't to be calculated off-line at the beginning. With the parameters offered by the parameter-regulating module, the cluster-analyzing module assorts the real data, sets up new clusters automatically, and regulates rules storeroom real time. Simultaneously, the parameter-regulating module amends the parameters of all clusters real time, and sends the results to the cluster analyzing module and the controlling module respectively. The phenomenon that new rules do not blend with the old ones is also eliminated. To prevent a too complex system, we set an upper limit to the number of new clusters. In this way, a real-time adaptive regulation of both the parameters and the system structure is accomplished with the available language information as well as the running data.

According to the analysis of performance of the MCFC stack, some initial rules are developed. Adopting the flow rate of fuel gas as the control, we try to keep the output voltage invariable while the temperature within its limit in case of disturbance of the external load.

Simulation results show that the proposed improved adaptive fuzzy controller is better than the traditional PID controller and the traditional fuzzy controller in the overtune, the stabilization rate, and the stable precision. The PID control is a trial technology disregarding the information of the plant. While the improved adaptive controller not only adopts the knowledge deduced from the plant but also the real-time data to regulate itself. Traditional fuzzy controller operates only according to the initial rules. It doesn't have the ability of self-regulation. The proposed adaptive fuzzy controller not only regulates the parameters of rules with the parameter-regulating module real time, but modifies the structure of the fuzzy system with the cluster analyzing module so as to make the system self-contained and reliable.

References

[1] Takagi H., Sugeno M. (1985), Fuzzy Identification of Systems and Its Applications to Modeling and Control, IEEE Trans. on Systems, Man. and Cybern., SMC-15(1), P116-132

[2] Wang L. X., Mendel J. M.(1992), Generating Fuzzy Rules from Numerical Data, IEEE Trans. on Systems, Man. and Cybern., 22(6):1414-1427

[3] Xu L., Kerzyzak A.(1993), Rival Penalized Competitive Learning for Clustering Analysis, IEEE Trans. NN., 4(4):636-649

[4] Zadeh L. A. (1965), Fuzzy Sets, Control, , 8, P338-353

[5] He Wei (1994), A Simulation Model for Integrated Molten Carbonate Fuel Cell Systems, Journal of Power Sources, 49:283-290

[6] Matsumoto S., Sasaki A., Urushibata H., et.al (1990), Performance Model of Molten Carbonate Fuel Cell, IEEE Transactions on Energy Conversion, 5(2):253-257

[7] Mugikura Y., Abe T., Watanabe T., et.al (1992), Analysis of Performance of Molten Carbonate Fuel Cell II . Development of a Performance Correlation Equation, Denki Kagaku, 60(2):124-130

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A novel genetic algorithm and its application in control of molten carbonate fuel cell

oXing-jin Sun, Guang-yi Cao, Xin-jian Zhu
Institute of Fuel Cell
Shanghai Jiaotong University
Shanghai 200030, P.R.China

Abstract

A novel genetic algorithm, ranking genetic algorithm is proposed. This algorithm includes two levels of structure: the families' level and the harmonizing level. The families are parallel during the process of evolution. The harmonizing level lines the families, and transports the best individual of the first-rank family to low-grade families so as to accelerate their evolution. Two levels of competition are constructed, one is among individuals of a family, and the other is among families. The competition in a family is accomplished by genetic algorithm with improved crossover operator. The mutation probability of a family is determined by its relative competition power. In this way, a fast and global convergence to the optimal goal is obtained. Crossover operator of GA is improved for the case of floating coding. The improved crossover operator can generate children individuals at random within the space of the super cube, which enhances the space searching rate and precision. Finally, the proposed novel GA is applied into the fuzzy-variable structure control system of molten carbonate fuel cell (MCFC). Simulation result is satisfying.

Keywords

RANKING GENETIC ALGORITHM - IMPROVED CROSSOVER OPERATOR - MOLTEN CARBONATE FUEL CELL (MCFC)

1 Introduction

Traditional GA adopts binary coding crossover operator. Michalewicz[1] pointed out that floating coding has higher precision and is convenient to deal with multi-variable systems as well as searching in large space. For floating coding, there are two kinds of crossover operators: linear crossover[1] and line crossover[2]. Both of them can enlarge the searching scope in a certain extent, while they are affected by coding coefficient. Standard crossover operator can only generate children individuals on the vertexes of the super cube formed by parent individuals. Children individuals generated by the linear crossover operator are on the diagonals of the super cube. It is the same for the line crossover operator but that the children individuals may be on the extensions of the diagonals.

Rudolph[3] pointed out that the global optimal result is not available for traditional GA. Eiben[4] put forward that the GA reserving the best individual can reach the global optimal result, while it may take a long time. Aiming at the deficiency of traditional GA, several improvements were done. For example, Goldberg[5] and Schraudolph[6] introduced the floating coding and dynamic parameter encoding (PDE). No mater in theory or application, there are still large amount of work for us in GA.

Molten carbonate fuel cell (MCFC) is an electric power generating equipment. To attain high performance and safety as well as long lifetime of the MCFC, its working temperature and gas pressure must be controlled, we will apply the proposed ranking GA with an improved crossover operator into the fuzzy variable structure control of MCFC.

2 Improved crossover operator

Traditional GA adopts binary coding. Michalowicz[1] put forward floating coding linear crossover operator,etc., numerical crossover operator. It has been applied to floating GA widely. It is shown in Eq.(1)

$$\begin{aligned} Y_1 &= \alpha X_1 + (1-\alpha)X_2 \\ Y_2 &= (1-\alpha)X_1 + \alpha X_2 \end{aligned} \qquad 0 \le \alpha \le 1 \quad (1)$$

where X_1 and X_2 are parent individuals, Y_1 and Y_2 are children individuals produced by linear crossover.

However, the linear crossover operator has a tendency to convergence to the middle of the searching space. To enlarge the searching space, Wright[7] advanced a linear crossover operator by which three children individuals were produced from two parent individuals. In fact, two of the three children individuals came from Eq.(1) where α=1.5. Rasheed[8] pointed out that children individuals may not be on the line between two parent individuals. They should be allowed to be on the extent of the line formed by the two parent individuals. This crossover operator is called line crossover, which makes α larger than one, namely, $0 \le \alpha$ in Eq.(1). The linear crossover and the line crossover are able to enlarge the searching scope in a certain extent, while they are affected by α in Eq.(1). For example, is α is close to one, then children individuals are similar with parent individuals, which means that the crossover is similar with the selection and the reproduction, and the searching efficiency is low. On the other hand, if α is much larger than one, then the colony will experience a danger of radiation.

Here we try to get rid of the opinion that the children individuals must be on the line formed by two parent individuals, and propose an improved linear crossover operator, $\vec{\alpha}$, whose elements are given between zero and one at random. For the function optimization problem with p variables, we have

$\vec{\alpha} = (\alpha_1, \alpha_2, \cdots \alpha_p)^T$, $\vec{X}_1 = (x_{11}, x_{12}, \cdots x_{1p})^T$, $\vec{X}_2 = (x_{21}, x_{22}, \cdots x_{2p})^T$, $\vec{Y}_1 = (y_{11}, y_{12}, \cdots y_{1p})^T$, and $\vec{Y}_2 = (y_{21}, y_{22}, \cdots y_{2p})^T$ respectively.

$$\vec{Y}_1 = \vec{\alpha} \times \vec{X}_1 + (\vec{1} - \vec{\alpha}) \times \vec{X}_2 = \begin{bmatrix} \alpha_1 x_{11} + (1-\alpha_1) x_{21} \\ \alpha_2 x_{12} + (1-\alpha_2) x_{22} \\ \vdots \\ \alpha_p x_{1p} + (1-\alpha_p) x_{2p} \end{bmatrix}$$

$$\vec{Y}_2 = (1 - \vec{\alpha}) \times \vec{X}_1 + \vec{\alpha} \times \vec{X}_2 = \begin{bmatrix} (1-\alpha_1) x_{11} + \alpha_1 x_{21} \\ (1-\alpha_2) x_{12} + \alpha_2 x_{22} \\ \vdots \\ (1-\alpha_p) x_{1p} + \alpha_p x_{2p} \end{bmatrix} \quad (2)$$

In fact, the optimal goal may not be on the line formed by parent individuals. The improved crossover operator can generate children individuals at random within the space of the super cube, which enhances the space searching rate and precision of crossover operator, especially for complex function optimization.

Now we will test the validity of the proposed improved crossover operator for optimazion of Rosenblock function (Eq.(3)) and Rastrigin function (Eq.(4))

$$\min f(X) = 100(x_1^2 - x_2)^2 + (1 - x_1)^2,$$
$$x_i \in [-2.048, 2.048], \quad i = 1,2 \quad (3)$$

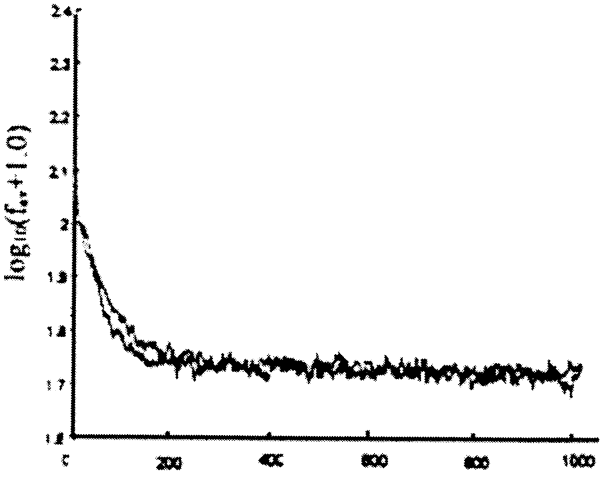
$$\min f(X) = \sum_{i=1}^{10} (x_i^2 - 10\cos(2\pi x_i) + 10),$$
$$x_i \in [-5.12, 5.12], \quad i = 1,2,\cdots 10 \quad (4)$$

To minimize other disturbances, we adopt a simple floating coding GA who reserves the best individual and changes the fitness value. The relative specifications are shown in Tab. 1.

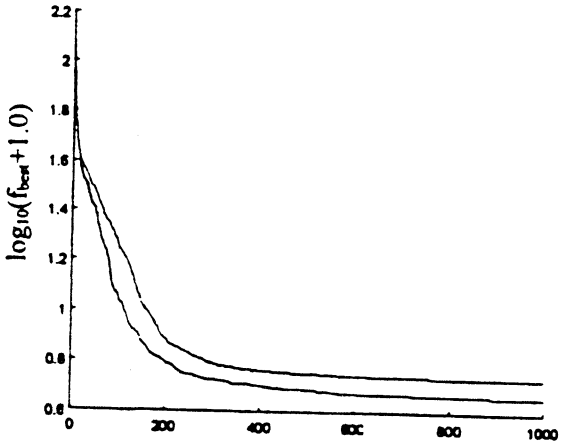
Table 1 Specifications of a simple floating coding GA

Item	Specification
Selection	Random
Mutaion	Gause $N(0,\sigma^2)$ $\sigma = 0.2$ (Rosenblock) $\sigma = 0.5$ (Rastrigin)
Crossover rate P_c	0.95
Mutation rate P_m	0.10
Population Size	50
Generation Number	300(Rosenblock) 1000(Rastrigin)

The improved crossover is compared with the linear crossover. To obtain a precise result, we will run the evolution independently for 30 times and take the average value for every problem and operator. For two-dimensional Rosenbrock function, the linear crossover and the improved linear crossover performance similarly; while for multi-dimensional Rastrigin function, the later shows faster optimization rate and higher precision, which is shown if Fig.1. The improved crossover performs well because it enlarges searching space for children individuals.



(a)



(b)

Fig.1 Optimization of Rastrigin function

3 Ranking genetic algorithm

Considering that there are two levels of competition in nature: one among families and the other among organisms of a family, we propose a novel genetic algorithm, ranking genetic algorithm. It includes two levels of structure: the families' level and the harmonizing level. The whole colony is divided into several sub-colonies, and each sub-colony is called a family. The number of families and that of individuals in each family are given at random. Two levels of competition are constructed according to those in nature. The competition in a family is accomplished by genetic algorithm with improved crossover operator. In each family, the individual having largest fitness stands for its family. The harmonizing level queues the families from the best to the worst according to their fitness, keeps the excellent families, and transports the best individual of the first-rank family to low-grade families so as to accelerate their evolution. The mutation probability of a family is determined by its relative competition power, namely, the distance of a family to the first-rank one in the queue. So the excellent families have little mutation probabilities, while the low-grade families have bigger ones. In this way, the first-rank family is substituted by one of the others, or the value of its fitness is refreshed constantly, which brings a fast convergence to the

optimal goal. At the same time, the families are independent and parallel during the process of evolution, which makes the chance of global convergence larger.

The procedure of the ranking GA is given below.

Step 1 Initialization. Generate individuals of every family at random.

Step 2 Calculate individuals' fitness values.

Step 3 Line the families based on the fitness value of each family's delegate.

Step 4 Calculate relative competition power as well as crossover rate and mutation rate.

Step 5 Each family experiences an evolution by selection, crossover and mutation. Transport the best individual of the first-rank family to low-grade families to accelerate their evolution.

Step 6 If ending condition is satisfied, go to (7), or else to (2).

Step 7 Output the result. Stop.

4 Application of the ranking GA in the control of the MCFC

We will establish a fuzzy-variable structure control system (FVSC) for the MCFC model, which is shown in Fig.3.

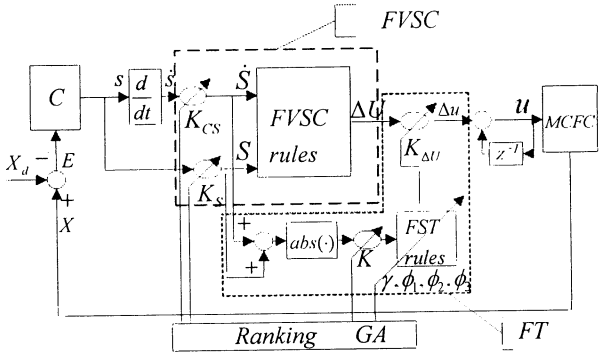


Fig.2 Scheme of FVSC system

The temperature model of the MCFC is given as Eq.(5)

$$\dot{X} = f(X) + g(X)u + \Delta f(X) \tag{5}$$

$$X_d = [923.07 \quad 943.1 \quad 943.1 \quad 951.8]$$

where $s = CE = C(X - X_d)$ $C = (c_1, c_2, c_3, 1)^T$

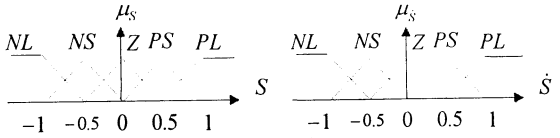
Fig.4 shows that

$$S(kT) = s(kT) \cdot K_S \quad \dot{S}(kT) = \dot{s}(kT) \cdot K_{CS} \tag{6}$$

$$\Delta u(kT + T) = \Delta U(kT + T) \cdot K_{\Delta U}$$

$$u(kT + T) = u(kT) + \Delta u(kT + T)$$

where T is sample cycle. Define fuzzy set of S, $\dot{S}$ and ΔU as { *NL*, *NS*, *Z*, *PS*, *PL* }, where *NL*, *NS*, *Z*, *PS* and *PL* denotes minus large, minus small, zero, positive small and positive large respectively. Their membership function are shown in Fig.4.



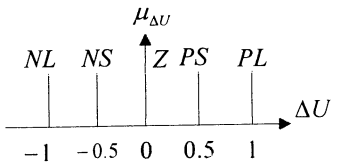


Fig.3 Membership functions μ_S、$\mu_{\dot{S}}$ and $\mu_{\Delta U}$

Fuzzy rules are designed based on the variable structure control technology, so as to ensure the stability and robustness of the control system. They are shown in Table.2.

Table 2 Linguistic rules of FVSC

ΔU		$\dot{S}$				
		NL	NS	Z	PS	PL
S	NL	PL	PL	PM	PS	Z
	NS	PL	PM	PS	Z	NS
	Z	PM	PS	Z	NS	NM
	PS	PS	Z	NS	NM	NL
	PL	Z	NS	NM	ML	NL

The scale factors of the fuzzy controller affect remarkably the response of the system. A fuzzy scale tuner (FST) is mounted to regulate FVSC scale factors.

Make $K \cdot |S + \dot{S}|$ the input variable of the FST, which is limited in the range of [0, 1], while the output variable $K_{\Delta U}$. The FST fuzzy rules are shown in Table 3. Membership functions of $K \cdot |S + \dot{S}|$ and $K_{\Delta U}$ are shown if Fig.5.

Table 3 Fuzzy rules of the fuzzy scale tuner (FST)

$K \cdot \lvert S + \dot{S} \rvert$	$K_{\Delta U}$
S	ϕ_1
M	ϕ_2
L	ϕ_3

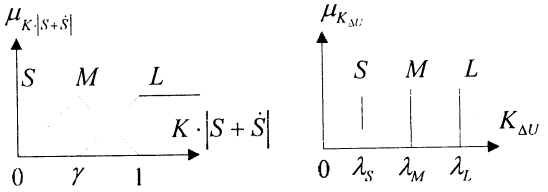


Fig.4 Membership functions of $K \cdot |S + \dot{S}|$ and $K_{\Delta U}$

To obtain an optimal FVSC and FT, we apply the ranking GA to search the optimal values of K_S, K_{CS}, K, γ、λ_S、λ_M and λ_L.

Select parameters of the ranking GA as shown in Table 4.

Table 4 Parameters of the ranking GA

Largest generation number	Population size	Crossover rate	Mutation rate
300	10	0.95	0.1~0.4

Resume there is a 5% step augment of the external load during operation, and we adopt the simulation results as the training data of the ranking GA. The parameters of the FVSC are shown in Table 5.

Table 5 Parameters of the FVSC with a FST

K_S	K_{CS}	K	γ	λ_S	λ_M	λ_L
0.05	.005	4.92	.301	0	45.03	97.15

5 Conclusion

We propose the ranking genetic algorithm. The excellent families have little mutation probabilities, while the low-grade families have large ones. In this way, the first-rank family is substituted by one of the others, or its fitness value is refreshed constantly, which brings a fast convergence to the optimal result. At the same time, the families are independent and parallel during the process of evolution, which enlarges the chance of global convergence.

We improved the crossover operator for the case of floating coding. We propose an improved linear crossover operator, which is able to generate children individuals at random within the space of the super cube, which enhances the searching rate and precision of the crossover operator, especially for complex function optimization. Furthermore, the function optimization simulation of the improved crossover operator compared with the others approves the analysis.

Finally, the proposed ranking GA with the improved crossover operator is applied into the control system of molten carbonate fuel cell (MCFC). A fuzzy-variable structure control system (FVSC) is designed. Fuzzy rules are designed based on the variable structure control technology, so as to ensure the stability and robustness of the control system. The global optimal parameters of the FVSC are searched by the proposed GA. Simulation results are satisfying.

References

[1] Michalewicz Z, Janikow C.Z., Krawczyk J.B., A modified genetic algorithm for optimal control problems. Computers Math. Applic., 1992,23(12):pp.83-94

[2] Rasheed K, Hirsh H, Gelsey A, A genetic algorithm for continuous design space search. Artificial intelligence in engineering,1997,11:pp.295-305

[3] Rudolph G., Convergence properties of canonical genetic algorithms. IEEE Trans. On Neural Networks, 1994,5(1):pp.96-101

[4] Eiben A.E., Aarts E.H., Van Hee K.M., Global convergence of genetic algorithms: An infinite Markov chain analysis. Proceedings of the 1st international Conference on Parallel Problem Solving from Nature, 1990,10

[5] Goldberg D.E., real-coded genetic algorithm, virtual alphabets and blocking, Complex systems,1991,5:pp.139-167

[6] Schraudolph N.N., Belew R.K. (1992), Dynamic parameter encoding for genetic algorithms. Machine learing, 9(6):9-21

[7] Wright A (1991), Genetic algorithms for real parameter optimization, In:Rawlins G.J.E.,(Ed.), foundations of genetic algorithms(FOGA'90), Morgan Kaufman, San Mateo CA, pp.205-218

[8] Rasheed K., Hirsh H, Gelsey A (1997), A genetic algorithm for continuous design space search. Artificial intelligence in engineering, 11:295-305

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A New Method for Designing Robust Neural Network Controller

Hongping Chen, Kotaro Hirasawa, Jinglu Hu and Junichi Murata
Intelligent Control Laboratory
Graduate School of Information Science and Electical Engineering, Kyushu University,
6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan

Abstract

In this paper, we propose a new method for designing a robust neural network controller against system environment changes based on the use of Universal Learning Networks (ULNs). A distinctive feature of the proposed method is that it searches concurrently the worst values of system parameters as well as the optimization of controller parameters, so that the robust controller can be obtained by minimizing the criterion function regarding the worst values of system parameters. Simulation shows that system performance has been improved with our method.

1 Introduction

Because of the ability of neural networks to approximate arbitrary nonlinear functions, there has been a growing interest in the nonlinear system control using neural networks. A neural network based controller can be designed in both on-line and off-line ways. A conventional off-line design method using neural networks is that the criterion function is usually evaluated at a specific operating point of system parameters. There is a problem in this conventional method, in which although the system performance at the evaluation point can be ensured, if system parameters at control stage are much different from those at training stage, performance of the control system becomes worse. Therefore, it is necessary to design a robust controller which not only performs well under the conditions it was trained, but also shows good generalization behavior to novel situations.

A method to solve this problem is the weighted multi-point evaluation method[1]. In a weighted multi-point evaluation method, the system is evaluated at multiple specific operating points of system parameters. The problem of this method is that it is difficult to choose the appropriate weights. Another way to solve this problem is the minimax control method[2] , in which the criterion function is evaluated at several specific operating points of the system parameters, and at each training step the worst criterion function among the operating points is optimized. By using the minimax control method, the criterion function at each special operating point can be ensured to be small. However, only limited evaluation points are usually practically chosen, and the system performance between the specific operating points may not be guaranteed.

In this paper, a new method of designing robust controllers for nonlinear systems using Universal Learning Network (ULN) is considered to overcome the above drawbacks of existing methods.

ULN[3, 4] is a superset of neural networks. It consists of two kinds of elements: nodes and branches. Nodes are assigned appropriate node functions including a function realized by a subnetwork, e.g. a sigmoidal neural network, and are interconnected with multi-branches that have arbitrary time delays. On the other hand, the system to be controlled can be expressed in a set of known equations derived from the physical law governing the system and the neural network controllers can be considered as a set of unknown equations parameterized using neural networks. Then, by corresponding nodes to the equations and branches to their relations, the equations describing the system and its controller form a learning network treated as a ULN. In this way, the controller design in a ULN-based control system is transformed into ULN learning.

In our proposed method, a dual learning algorithm is introduced for the ULN learning. It includes two searching procedures: *maximization* and *minimization*. In the maximization procedure, specific system parameters are adjusted so that the criterion function gives the worst value; while in the minimization procedure, controller parameters are optimized by minimizing the criterion function. In this way, the worst criterion function with respect to the system parameters can be minimized, then the robustness of the control system could be improved even though the system parameters are changed at control stage from those at

training stage.

2 The Proposed Method

In order to obtained a controller that is robust against the changes of system parameters, we propose a dual learning algorithm for the ULN learning, which includes two searching procedures: *maximization* and *minimization*. In concrete, the learning is carried out alternatively according to the following two steps: Minimize the criterion function to obtain the optimal controller parameters, while maximize the criterion function to obtain the system parameters which are changed in the admissible range.

Let $\lambda = \{\lambda_1, \ldots, \lambda_m, \ldots, \lambda_M\}$ denote the controller parameters, $f = \{f_1, \ldots, f_g, \ldots, f_G\}$ the system parameters , $E(\lambda, f)$ the criterion function, and $f_g \in [\underline{f_g}, \overline{f_g}]$ the system parameter which may change at control stage from the one at training stage, respectively. Then, the ULN learning can be described as the following dual optimization problem,

$$\lambda^\circ = \arg\min_{\lambda^\circ} E(\lambda, f) \quad (1)$$

$$f^\circ = \arg\max_{f^\circ} E(\lambda, f) \quad (2)$$

Therefore, through the above two procedures, the optimal controller can be obtained by minimizing the worst value of the criterion function with respect to the system parameters as follows,

$$\lambda^\circ = \arg\min_{\lambda^\circ}\{\max_{f^\circ} E(\lambda, f)\} \quad (3)$$

To solve the optimization problem (1) and (2), we introduce a dual learning algorithm as follows,

- Step 1) Choose the initial values of the system parameter f and the controller parameter λ. Set training time $t = 0$.
- Step 2) Minimizing the criterion function $E(\lambda, f)$ with respect to λ with f fixed.
- Step 3) Maximizing the criterion $E(\lambda, f)$ with respect to f with λ fixed.
- Step 4) If the end condition is satisfied, STOP, otherwise, set $t \to t+1$ and go to Step 2.

In this paper, the gradient method is used for learning controller parameters λ_m which is realized by minimizing the criterion function E and for searching the worst values of system parameters by maximizing the criterion function as follows,

$$\lambda_m \longleftarrow \lambda_m - \gamma \frac{\partial^\dagger E}{\partial \lambda_m} \quad (4)$$

$$f_g \longleftarrow f_g + \beta \frac{\partial^\dagger E}{\partial f_g} \quad (5)$$

where $\frac{\partial^\dagger E}{\partial \lambda_m}$ and $\frac{\partial^\dagger E}{\partial f_g}$ are the ordered derivatives defined by Werbos [5]which means the change of the criterion function E caused by the change of λ_m or f_g with other variables being fixed, and γ and β are the learning coefficients. It is can be seen that the proposed method is characterized by searching concurrently for the most worst values of system parameters as well as the optimization of controller parameters. After two searching procedures are carried out alternatively, the worst criterion function is minimized, therefore the robust controller can be obtained.

3 Simulation

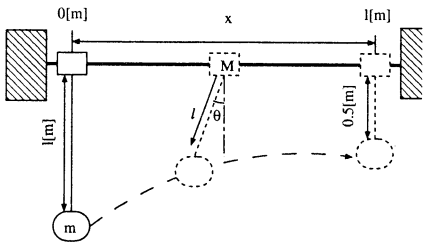


Figure 1: Structure of nonlinear crane system

The controlled system is a nonlinear crane system shown in **Fig.1**. The position of the crane stand, the angle between the rope and the vertical line, and the position of the load are denoted by x, θ, and l, respectively. Then the nonlinear crane system is described as follows,

$$\frac{d^2x}{dt^2} = -\frac{mg}{M}\theta - \frac{D+G}{M}\frac{dx}{dt} + \frac{G}{lM}u_1 \quad (6)$$

$$\frac{d^2\theta}{dt^2} = -\frac{M+m}{lM}g\theta - \frac{D+G}{lM}\frac{dx}{dt} + \frac{G}{lM}u_1 \quad (7)$$

$$\frac{d^2l}{dt^2} = -\frac{C+G_m}{m}\frac{dl}{dt} + \frac{G_m}{m}u_2 \quad (8)$$

where, u_1 and u_2 are input voltages for moving the crane stand and rolling up the load, respectively, and C, g, G_m, D, M and m are appropriate system parameters. Eq.(6), (7), and (8) are transformed to the

discrete time forms:

$$h_1(t) = h_1(t-1) + \Delta t \cdot h_2(t-1) \quad (9)$$

$$h_2(t) = \left(1 - \Delta t \frac{D+G}{M}\right) h_2(t-1) - \frac{mg}{M}\Delta t \cdot h_3(t-1) + \frac{G}{M}\Delta t \cdot u_1(t-1) \quad (10)$$

$$h_3(t) = h_4(t-1) + \Delta t \cdot h_4(t-1) \quad (11)$$

$$h_4(t) = h_4(t-1) - \frac{D+G}{M}\Delta t \frac{h_2(t-1)}{h_5(t-1)} - \frac{M+m}{M} g \Delta t \frac{h_3(t-1)}{h_5(t-1)} + \frac{G}{M}\Delta t \frac{u_1(t-1)}{h_5(t-1)} \quad (12)$$

$$h_5(t) = h_5(t-1) + \Delta t \cdot h_6(t-1) \quad (13)$$

$$h_6(t) = \left(1 - \frac{C+G_m}{m}\Delta t\right) h_6(t-1) + \frac{G_m}{m}\Delta t \cdot u_2(t-1) \quad (14)$$

where

$$h_1(t) = x(t) \quad , \quad h_3(t) = \theta(t) \quad , \quad h_5(t) = l(t)$$
$$h_2(t) = \dot{x}(t) \quad , \quad h_4(t) = \dot{\theta}(t) \quad , \quad h_6(t) = \dot{l}(t)$$

The nonlinear crane system and its controller can be

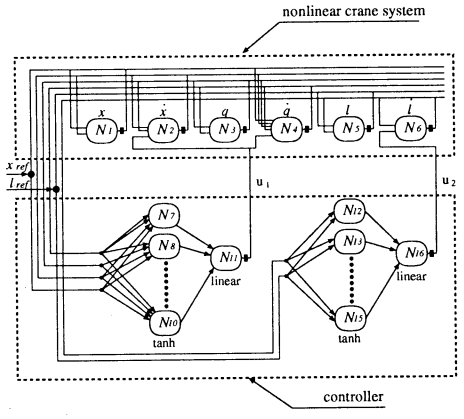


Figure 2: A feedback control system of nonlinear crane system using a layered controller

represented by a single ULN as shown in **Fig.2**. Each of the control inputs u_1 and u_2 is composed of five control nodes; one is the node with a linear function, the other four nodes are the nodes with tanh function. All the branches have one-sampling time delays. The aim of the control are shown as follow,

the crane stand position : $0 \rightarrow 1.0$ [m]
the load height : $1 \rightarrow 0.5$ [m]

The criterion function was set as:

$$E = \frac{1}{2}\Big[\sum_{s\in T}\{Q_1(x_{ref} - x(s))^2\} + Q_2\dot{x}(t_f)^2 + \sum_{s\in T}\{Q_3\theta(s)^2 + Q_4\dot{\theta}(s)^2\} + \sum_{s\in T}\{Q_5(l_{ref} - l(s))^2\} + Q_6\dot{l}(t_f)^2 + \sum_{s\in T}\{R_1u_1(s)^2 + R_2u_2(s)^2\}\Big] \quad (15)$$

t_f : final control instant(=7.5[sec])
T : set of sampling instants(=2500))

In these simulations, we take the mass of the load m and the coefficient for transforming voltage to torque G_m as the system parameters f which change at control stage from the those at training stage. The aim of the crane control system is to design a controller so that it gives better performances even though the mass of the load m and the coefficient for transforming voltage to torque G_m change in a wide range.

First, we carried out a simulation by using the one point evaluation method. The criterion function is evaluated at the point of $m = 2$ $[kg]$ $G_m = 1.5$ $[N/V]$, and then we trained the controller. At control stage, system responses were calculated at the 9 operating points as shown in **Table.1**, The responses with respect to l were shown in **Fig.3**. From this figure, we can see the performance at the operating point $m = 2$ $[kg]$ $G_m = 1.5$ $[N/V]$ is the best, but the more the value of m and G_m at control stage differ from those at the training stage, the worse the system performance is. Next, we did a simulation using the proposed method. We assume that the mass of the load may be changed between 2 kg and 8 kg, and the coefficient for transforming voltage to torque G_m may be changed between 0.4 $[N/V]$ and 1.5 $[N/V]$, then we carried out a dual learning algorithm described in Section 2. After training, system responses with respect to l were shown in **Fig.4** with the obtained controller. From this figure, it is clear that the system performance is improved with our proposed method compared with the conventional method. In order to show

Table 1: system parameters $f = \{m, G_m\}$ used at the control stage

	1	2	3	4	5	6	7	8	9
m	2	2	2	5	5	5	8	8	8
Gm	0.4	1.0	1.5	0.4	1.0	1.5	0.4	1.0	1.5

the generalization capability of our proposed method, we changed the initial position of the load and the reference of the load height as follows,

the crane stand position : $0 \rightarrow 1.5$ [m]
the load height : $1.5 \rightarrow 0.5$ [m]

And then system responses with respect to l were obtained as shown in **Fig.5** using the optimized controller. It can be seen that a level of performance can be guaranteed even though the initial values and the reference values of system have changed because of the generalization ability of the neural networks.

4 Conclusions

In this paper, we proposed a new design method of nonlinear control systems based ULNs, in which the worst values of the criterion function with respect to system parameters are searched as well as the optimization of controller parameters. Through the simulations of a nonlinear crane system, it can be seen that the proposed method is very efficient for designing the robust controller of a nonlinear system where the dynamics of the system are known and differentiable.

References

[1] M. Obayashi, K. Hirasawa, N. Toshimitsu, J. Murata and J. Hu, "Robust Control for Universal Learning Network Considering Fuzzy Criterion and Second Order Derivatives", *Trans. of SICE*, Vol. 34, No. 9, pp.1246-1254, 1998 (in Japanese).

[2] H. Chen, K. Hirasawa, J. Hu, J. Murata, " Min Max Control of Nonlinear Systems Using Universal Learning Networks", in Proc. *IEEE Int. Joint Conf. on Neural Networks.*, Vol.I, pp.242-247, 2000.

[3] K. Hirasawa, X. Wang , J. Murata, J. Hu, C. Jin, "Universal Learning Network and its application to chaos control", *Neural Networks*, Vol.13, pp.239-253, 2000.

[4] K. Hirasawa, J. Murata, J. Hu, C. Jin, "Universal Learning Networks and Its Application to Robust Control", *IEEE Trans. Syst., Man, and Cybern., part B*, Vol. 30, No. 3, pp.419-430, June 2000.

[5] P.Werbos; Beyond Regression, "New Tools for Prediction and Analysis in the Behavioral Sciences", Ph.D.dissertation, Harvard University, 1974.

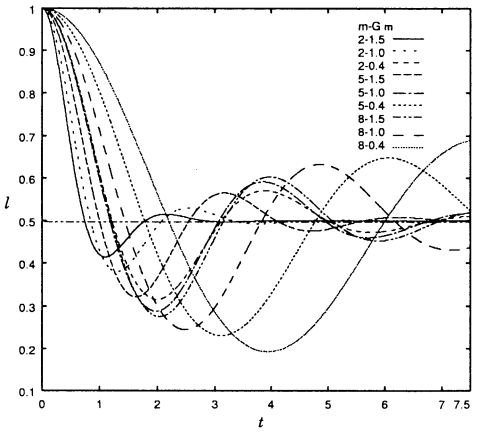


Figure 3: Responses of one point evaluation method

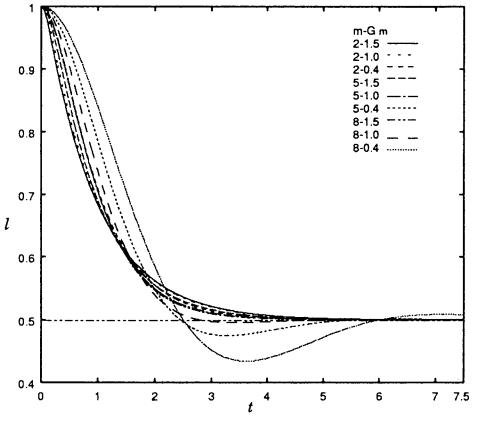


Figure 4: Responses of the proposed method

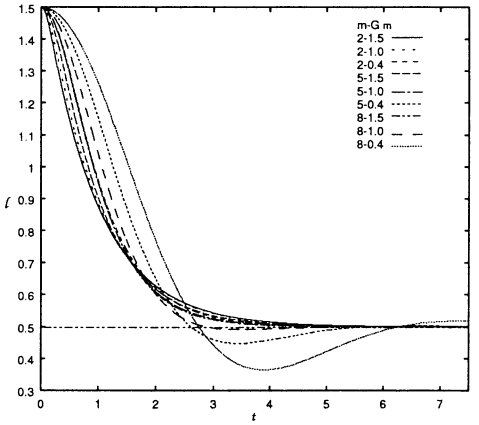


Figure 5: Generalization of the proposed method

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Modeling and Evolutionary Learning of Modular Neural Networks

Qiangfu Zhao

The University of Aizu
Aizu-Wakamatsu, Fukushima 965-8580, Japan
Email: qf-zhao@u-aizu.ac.jp

Abstract

In the last decade, a number of neural network models have been proposed in the literature. Some of them have been successfully incorporated in different intelligent information processing systems. Among these models, a group of most successful ones are the modular neural networks (MNNs). This paper introduces a general model of MNNs, and proposes a neural network tree (NNTree) model. An evolutionary algorithm is also given for designing the NNTrees. The usefulness of the NNTrees and the effectiveness of the learning algorithm are verified through experiments with a digit recognition problem.

1 Introduction

In the last decade, a number of neural network models have been proposed in the literature. Some of them have been successfully incorporated in different intelligent information processing systems [1]. Among these models, a group of most successful ones are the modular neural networks (MNNs) [2]. Many neurophysiology evidences support the modularity of the human brain — the true neural network model we want to build. Modularity is also useful from an engineering point of view. For example, MNNs are usually more scalable for solving large scale problems, suitable for parallel implementation, flexible in changing environments, and so on.

Even talking about the MNNs alone, many models have been proposed and studied separately. In this paper, we introduce a general framework for studying MNNs, and propose a new MNN model called neural network tree (NNTree). An evolutionary algorithm is also given for designing the NNTrees. The usefulness of the NNTrees and the effectiveness of the learning algorithm will be verified through experiments with a digit recognition problem.

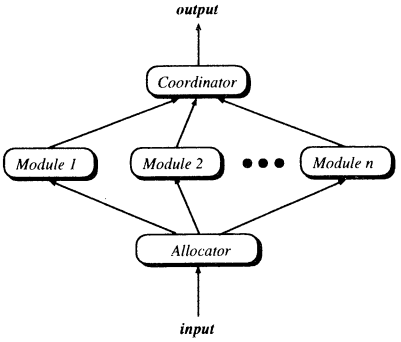


Figure 1: A general model for modular neural networks

2 A General Model for MNNs

Fig. 1 is a general model for MNNs. First, let us see how it works. When an input task is given, the **allocator** determines which module(s) should be used to fulfill this task. Generally, many modules might be selected to fulfill the task together. Each of these selected modules outputs a result based on local computation. The **coordinator** then gives the final result based on outputs of the modules. If the allocator is so strong that a single module can always be correctly selected to perform a given task, the coordinator can be removed. If, on the other hand, the allocator is so weak that all modules must be used to fulfill a task, a strong coordinator would be useful to make the final judgment. Interesting enough, most existing MNNs are different from each other simply because their al-

locators or coordinators are stronger or weaker. Here are some examples:

1. **The Classification and Regression Tree**
   A classification and regression tree (CART) [3] can be considered as a MNN with a very strong allocator which is the tree itself. The allocator is so strong that the final result can be given straightforwardly based only on examples assigned by the tree to one of the terminal nodes. In this case, both the coordinator and the modules are so simple that we can hardly say that CART is also a *neural* network.

2. **The Fuzzy Decision Trees**
   In a fuzzy decision or regression tree (FDT) [6], the allocator makes only fuzzy or *soft* splits (vs. the crisp splits used in CART), and therefore the final output depends on results of all terminal nodes. Thus, a coordinator should be used to integrate the results of all terminal nodes.

3. **The Hierarchical Mixture of Experts**
   The hierarchical mixture of experts (HME) given in [9] is a well know MNN model. A HME can also be considered as a decision tree which performs soft splitting [10]. In a HME, the gating networks are used as a coordinator who knows roughly how tasks are allocated. As in fuzzy decision trees, the allocator in a HME also performs soft splitting, but the splitting process stops much earlier. Therefore, stronger modules (experts) should be used to make a compensation.

4. **Neural Network Ensemble (NNE)**
   If we use a very weak allocator, or simply delete the allocator, we have the neural network ensemble (NNE). Usually, each module is a neural network designed using

   - different feature set,
   - different training set,
   - different initial condition,
   - different training algorithms,

   and so on [11]-[16]. After designing each module separately, they are integrated together using a coordinator. What a coordinator does is very simple: just averaging the results of all modules usually provides a good final result.

5. **One-Class-One-Network**
   If the modules are strong enough to *know* what to do, both the allocator and the coordinator can be deleted. The one-class-one-network (OCON) model is an example [17]. This is an extreme case of MNNs.

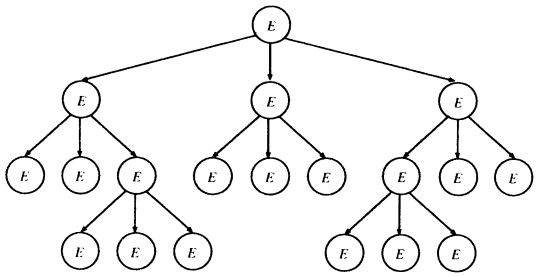


Figure 2: Neural network tree — another type of modular neural network

3 The Neural Network Tree

Fig. 2 is a neural network tree (NNTree). In this model, each node is an *expert neural network* (an E-module). All modules have the same complexity. In this paper, each E-module is a multilayer perceptron (MLP) with the same number of layers and the same number of neurons in each layer. The NNTree works as follows. The input is given to the root node (the first E-module) first. It is then assigned to the i-th child if the i-th output of the module is the maximum. If the child is a leaf, the final result is produced locally; otherwise, repeat the same procedure recursively.

The most notable feature of an NNTree is that it consists of homogeneous neural networks that can be realized using exactly the same functional components (with different parameters), and the whole system can be constructed hierarchically. In designing the system, we can use a similar procedure used in designing a CART. The only difference is that, instead of a simple comparison, an MLP is used in each node. Compared with the general MNN model given in the last section, we can see that in an NNTree, the allocator is the tree itself, and there is no coordinator. Each E-module can also be considered as a local allocator as well as a local decision maker.

The idea of using neural networks to extract features was first proposed in [18]. In the model given in [18], a non-terminal node is used only for extracting a single feature. In our model, however, a node (non-terminal or terminal) can extract many features, and make local decision based on the feature(s).

It is interesting to note that from an NNTree, we

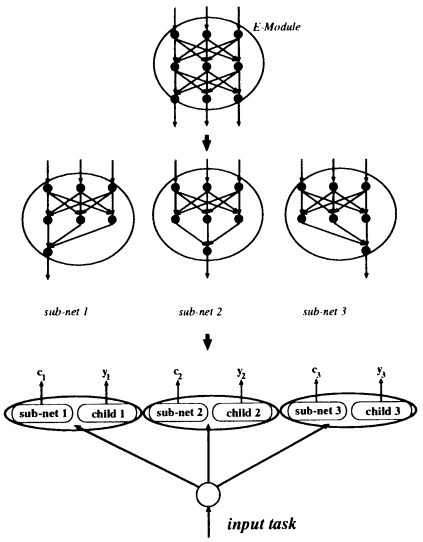


Figure 3: Mapping an NNTree to an AMNN

can easily get an *autonomous* modular neural network (AMNN). For example, if the first E-module (the root) is an MLP with n outputs, it can be split into n sub-nets. These sub-nets can then be used to work together with the children (see Fig. 3 for n=3). The AMNN works like this: for a given input task, each module tries to give an output y along with a *certainty number* c. If c_i is the maximum, the output of the $i-th$ module will be used as the final result. That is, the decision made by the agent who speaks most loudly will be accepted. Note that each module can be an AMNN again which can be obtained from the NNTree by using the same process recursively.

4 Evolutionary Learning of NNTrees

To construct a decision tree, it is often assumed that a training set consisting of feature vectors and their corresponding class labels are available. The decision tree is then constructed by partitioning the feature space in such a way as to recursively generate the tree. This procedure involves three steps: splitting nodes, determining which nodes are terminal nodes, and assigning class labels to terminal nodes. Among them, the most important and most time consuming step is splitting the nodes.

One of the popular algorithms for designing decision trees is C4.5 [5]. In C4.5, the *information gain ratio* is used as the criterion for splitting nodes. The basic idea is to partition the current training set in such a way that the average information required to classify a given example can be reduced most (for detailed information, see [5]).

Table 1: Experimental results - Part I

	Error-1	Error-2	Size
NNTree	0	163	84
Best	0	133	73
Min	0	182	69
DT (before pruning)	76	257	335
DT (after pruning)	81	256	319

Now, let's see how to design an NNTree. The overall process is the same as that used by C4.5. The key problem left is how to find an E-module for each node to maximize the information gain ratio. To solve this problem, we adopt a simple genetic algorithm (SGA) which has three operators: truncate selection, one-point crossover, and bit-by-bit mutation. We use this SGA just because it is simple. The genotype of an E-module is the concatenation of all weight vectors (including the threshold values) represented by binary numbers. The fitness is defined directly as the information gain ratio.

5 Experimental Results

To test the effectiveness of the algorithm proposed above, as well as the usefulness of the NNTrees, we conducted some experiments with a digit recognition problem. The data used here is the "optdigits" data set, which is taken from machine learning repository of the University of California at Irvine. The number of training examples is 3823, and the number of test examples is 1797. The number of features is 64 with each feature being an integer in [0,16]. The number of class is 10 (10 digits). Detailed information can be found in the file optdigits.names, which is included in the data set.

The main experiment parameters are as follows. First, for each E-module, the number of output neurons is 2, the number of hidden neurons is 4, and the number of bits per weight is 16. The number of generations is 1000, The population is 200, the selection rate is 0.2, the crossover rate is 0.7, and the mutation rate is 0.01.

Table 1 shows the experimental results. In this table, Error-1 is the number of classification errors for the training data, Error-2 is that for the test data, and

Size is the number of nodes (including non-terminal and terminal nodes) of the Tree. The first raw is the average result over 10 runs, the second raw is the best result in 10 runs, and the third raw is the minimum NNTree obtained in 10 runs. Compared with the results obtained by C4.5 (the 4-th and the 5-th raws) it is clear that the number of nodes has been greatly reduced, and at the same time, the number of classification errors is also much smaller.

6 Conclusion and Remarks

In this paper, we have introduced a new type of modular neural networks — neural network tree (NNTree), and proposed an evolutionary training algorithm. Experimental results with a digit recognition problem show that NNTree is more efficient than traditional decision trees in the sense that higher recognition rate can be achieved with less nodes.

There are still many open problems to be solved in the future. For example, how to train the E-modules on-line, and how to interpret the E-modules as well as the whole network ? All these are interesting topics for further study.

References

[1] S. Y. Kung and J. N. Hwang, "Neural networks for intelligent multimedia processing," Proc. IEEE, Vol. 86, pp. 1244-1272, June, 1998.

[2] T. Caelli, L. Guan and W. Wen, "Modularity in neural computing," Proc. IEEE, Vol. 87, No. 9, pp. 1497-1518, 1999.

[3] L. Brieman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "Classification and regression trees," Belmont, CA: Wadsworth, 1984.

[4] J. R. Quinlan, "Induction of decision trees," Machine Learning, Vol. 1, No. 1, pp. 81-106, 1986.

[5] J. R. Quinlan, "C4.5: Programs for machine learning," Morgan Kaufmann Publishers, 1993.

[6] C. Z. Janickow, "Fuzzy decision trees: issues and methods," IEEE Trans. on Systems, Man, and Cybernetics B, Vol. 28, No. 1, pp. 1-14, 1998.

[7] A. Suárez and J. Lutsko, "Globally optimal fuzzy decision trees for classification and regression," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 21, No. 12, pp. 1297-1311, 1999.

[8] R. A. Jacobs and M. I. Jordan, "Adaptive mixtures of local experts," Neural Computation, Vol. 3, pp. 79-87, 1991.

[9] M. I. Jordan and R. A. Jacobs, "Hierarchical mixtures of experts and EM algorithm," Neural Computation, Vol. 6, pp. 181-214, 1994.

[10] S. Waterhouse and T. Robinson, "Constructive methods for mixtures of experts," NIPS: Neural Information Processing Systems, 1995.

[11] L. K. Hansen and P. Salamon, "Neural network ensembles," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 12, No. 10, pp. 993-1001, 1990.

[12] G. Rogova, "Combining the results of several neural network classifiers," Neural Networks, Vol. 7, No. 5, pp. 777-781, 1994.

[13] B. L. M. Happel and J. M. J. Murre, "Design and evolution of modular neural network architectures," Neural Networks, Vol. 7, No. 6/7, pp. 985-1004, 1994.

[14] H. Drucker, C. Cortes, L. D. Jackel, Y. LeCun and V. Vapnik, "Boosting and other ensemble methods," Neural Computation, Vol. 6, pp. 1289-1301, 1994.

[15] S. B. Cho and J. K. Kim, "Multiple network fusion using fuzzy logic," IEEE Trans. on Neural Networks, Vol. 6, No. 2, pp. 497-501, 1995.

[16] X. Yao and Y. Liu, "Making use of population information in evolutionary artificial neural networks," IEEE Trans. on Systems, Man, and Cybernetics - Part B, Vol. 28, No. 3, pp. 417-425, 1998.

[17] S. Y. Kung and J. S. Taur, "Decision-based neural networks with signal/image classification applications," IEEE Trans. on Neural Networks, Vol. 6, No. 1, pp. 170-181, 1995.

[18] H. Guo and S. B. Gelfand, "Classification trees with neural network feature extraction," IEEE Trans. on Neural Networks, Vol. 3, No. 6, pp. 923-933, 1992.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

H_∞-Control for Fuzzy Time-Delay Systems

Jun Yoneyama
Department of Electronics and Electrical Engineering,
Aoyama Gakuin University,
6-16-1 Chitosedai, Setagaya, Tokyo 157-8572 Japan
E-mail: yoneyama@ee.aoyama.ac.jp,
Phone and Fax: +81 3 5384 6418

Abstract

In this paper the design problem of both H_∞ state and output feedback controllers for Takagi-Sugeno fuzzy time-delay systems is considered. We first introduce Takagi-Sugeno fuzzy time-delay systems and the H_∞-performance (norm) for a stable fuzzy time-delay systems. We give a sufficient condition for the norm being less tham a given number. We then consider the H_∞-problem with state and output feedback controllers. In the design of output feedback controllers, the selection of the premise variables in the if-then rules is important. We assume that the premise variable is given. In this case, we give design methods of H_∞-controllers and sufficient conditions which guarantee the required H_∞-performance of the closed-loop system.

Keywords
H_∞-control; Takagi-Sugeno fuzzy time-delay systems; Riccati inequalities

1 Introduction

The Takagi-Sugeno fuzzy dynamic model is a system described by fuzzy if-then rules which give local linear representations of the underlying nonlinear system (Takagi and Sugeno [1], Tanaka and Sugeno [2]). It is well-known as in Tanaka et al. [3] that such a model can describe or approximate a wide class of nonlinear systems. Since the work by Tanaka and Sugeno in [2] on stability analysis and state feedback stabilization there has been much development in system theory for such systems. For fuzzy systems H_∞-control has been considered by many authors. The bounded real lemma is formulated in terms of LMI's in Hong and Langari [4] and Tanaka et al. [3], the design of state feedback H_∞-controllers is considered in Cao et al. [5], Feng et al. [6], and the design of output feedback H_∞-controllers, which are more important and practical, has appeared in Yoneyama et al. [7] where a design method of output feedback H_∞-controllers based on the two Riccati inequalities of standard linear systems has been discussed and sufficient conditions for the proposed controller to be γ-suboptimal have been given.

In this paper, we consider the H_∞-control problem for fuzzy time-delay systems. Although stabilizing controllers for systems with time-delays have been obtained in Delfour [8] and Gibson [9], it is difficult to implement such controllers in practice due to the infinite dimension of the controllers. Recently, it has been shown in Kubo and Shimemura [10] that memoryless feedback controllers can stabilize linear systems with state time-delays. We shall show that such controllers can also be γ-suboptimal controllers for fuzzy time-delay systems. We first give sufficient conditions for the H_∞-norm of a stable fuzzy time-delay system to be less than a given number. We then consider the H_∞-control problem with state feedback and give sufficient conditions for a controller to be γ-suboptimal. Finally, we give a design method of H_∞-output feedback controllers.

2 H_∞-Control

We consider H_∞ state and output feedback controllers for Takagi-Sugeno fuzzy time-delay models, and give sufficient conditions for existence of the H_∞-controllers.

2.1 H_∞-Performance

In this section, we introduce the fuzzy time-delay model of Takagi and Sugeno and give H_∞-optimality conditions for the homogeneous models.

Consider the Takagi-Sugeno fuzzy time-delay model described by the following fuzzy IF-THEN rules:

$$\begin{array}{ll} \textbf{IF} & \xi_1 \text{ is } M_{i1} \text{ and } \cdots \text{ and } \xi_q \text{ is } M_{iq}, \\ \textbf{THEN} & \dot{x}(t) = A_i x(t) + E_i x(t-h) + B_i w(t), \\ & z(t) = C_i x(t), \quad i = 1, \cdots, r \end{array}$$

where $x \in \mathcal{R}^n$ is the state, $w \in \mathcal{R}^{m_1}$ is the disturbance, $z \in \mathcal{R}^{p_1}$ is the controlled output, the matrices A_i, B_i, C_i and E_i are of appropriate dimensions, r is the number of IF-THEN rules, M_{ij} are fuzzy sets and

$\xi = [\xi_1 \ \cdots \ \xi_q]^T$ is the premise variable. Here we assume that the premise variable is given. The state equation and the controlled output are defined as follows:

$$\begin{aligned}\dot{x}(t) &= \sum_{i=1}^{r} \lambda_i(\xi(t))\{A_i x(t) + E_i x(t-h) + B_i w(t)\},\\ z(t) &= \sum_{i=1}^{r} \lambda_i(\xi(t)) C_i x(t).\end{aligned} \tag{1}$$

where

$$\lambda_i(\xi) = \frac{\beta_i(\xi)}{\sum_{i=1}^{r}\beta_i(\xi)}, \quad \beta_i(\xi) = \prod_{j=1}^{q} M_{ij}(\xi_j)$$

and $M_{ij}(\cdot)$ is the grade of the membership function of M_{ij}. We assume $\beta_i(\xi) \geq 0$, $i = 1, \cdots, r$, and $\sum_{i=1}^{r}\beta_i(\xi) > 0$, for any t. Hence $\lambda_i(\xi)$ satisfy $\lambda_i(\xi) \geq 0$, $i = 1, \cdots, r$ and $\sum_{i=1}^{r}\lambda_i(\xi) = 1$ for any t.

Definition 2. 1 *The fuzzy system desribed by (1) is input-output stable if $z(\cdot) \in L^2(0,\infty;\mathcal{R}^{p_1})$ for any $w(\cdot) \in L^2(0,\infty;\mathcal{R}^{m_1})$ where $L^2(0,\infty;\cdot)$ is the space of square integrable functions.*

We wish to find sufficient conditions that the system (1) is exponentially stable and input-output stable with

$$||z||_2 \leq d||w||_2 \text{ for some } 0 < d < \gamma \tag{2}$$

where $\|\cdot\|_2$ is the L^2-norm. In this case we say that the H_∞-norm of the system (1) is less than γ. For the usual linear systems we have the following lemma.

Lemma 2. 1 *Consider the following system*

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Ex(t-h) + Bw(t),\\ z(t) &= Cx(t).\end{aligned}$$

Then it is exponentially stable and (2) is satisfied if and only if there exist $X_1 > 0$ and $X_2 > 0$ which satisfy

$$\begin{aligned}&A^T X_1 + X_1 A + X_2 + C^T C\\ &\quad + X_1(\frac{1}{\gamma^2}BB^T + EX_2^{-1}E^T)X_1 = -P < 0\end{aligned}$$

for some $P > 0$.

We generalize the sufficiency part to the system (1).

Proposition 2. 1 *Suppose there exist common matrices $X_1 > 0$ and $X_2 > 0$ such that*

$$\begin{aligned}&A_i^T X_1 + X_1 A_i + X_2 + C_i^T C_i\\ &\quad + X_1(\frac{1}{\gamma^2}B_i B_i^T + E_i X_2^{-1} E_i^T)X_1\\ &\qquad = -P_i < 0, \ i = 1, \cdots, r,\end{aligned} \tag{3}$$

for some $P_i > 0$. Then the fuzzy system (1) is stable and input-output stable with (2).

2.2 H_∞-Control with State Feedback

First we consider the H_∞-control problem with state feedback for the Takagi-Sugeno fuzzy time-delay models. In this case a natural assumption in the premise variales is $\xi = x$ ([3]). But to keep the generality, we will use $\xi = [\xi_1 \ \cdots \ \xi_l]^T$ as the premise variables. Consider the following fuzzy rules:

$$\begin{aligned}&\text{IF} \quad \xi_1 \text{ is } M_{1i} \text{ and } \cdots \text{ and } \xi_p \text{ is } M_{pi},\\ &\text{THEN} \quad \dot{x}(t) = A_i x(t) + E_i x(t-h)\\ &\qquad\qquad + B_{1i} w(t) + B_{2i} u(t),\\ &\qquad\qquad z(t) = \begin{bmatrix} C_i x(t) \\ D u(t) \end{bmatrix}, \ i = 1, \cdots, r\end{aligned} \tag{4}$$

where $x(t) \in \Re^n$, $w(t) \in \Re^q$, $u(t) \in \Re^m$, $z(t) \in \Re^{l_1+l_2}$, $C_i \in \Re^{l_1 \times n}$, $D \in \Re^{l_2 \times m}$ and other matrices are of appropriate dimension. We assume $D^T D = I$ (We may assume $l_2 = m$ and $D = I$.). The state and controlled output equations are defined as follows;

$$\begin{aligned}\dot{x}(t) &= \sum_{i=1}^{r} \lambda_i(\xi(t))(A_i x(t) + E_i x(t-h)\\ &\qquad + B_{1i} w(t) + B_{2i} u(t)),\\ z(t) &= \sum_{i=1}^{r} \lambda_i(\xi(t)) \begin{bmatrix} C_i x(t) \\ D u(t) \end{bmatrix}.\end{aligned} \tag{5}$$

For each subsystem in (4), we assume the following rules are given;

$$\begin{aligned}&\text{IF} \quad \xi_1 \text{ is } M_{1i} \text{ and } \cdots \text{ and } \xi_p \text{ is } M_{pi},\\ &\text{THEN} \quad u(t) = F_i x(t), \ i = 1, \cdots, r.\end{aligned} \tag{6}$$

Then the natural choice of the controller is given by

$$u(t) = \sum_{i=1}^{r} \lambda_i(\xi(t)) F_i x(t). \tag{7}$$

We use the same weights $\lambda_i(\xi(t))$ as those for the rules (4). Substituting the controller (7) into the fuzzy time-delay system (5), we obtain the closed-loop system

$$\begin{aligned}\dot{x}(t) = &\sum_{i=1}^{r}\sum_{j=1}^{r} \lambda_i(\xi(t))\lambda_j(\xi(t))(A_i + B_{2i}F_j)x(t)\\ &+ \sum_{i=1}^{r} \lambda_i(\xi(t))(E_i x(t-h) + B_{1i} w(t)),\\ z(t) = &\sum_{i=1}^{r} \lambda_i(\xi(t)) \begin{bmatrix} C_i \\ DF_i \end{bmatrix} x(t).\end{aligned} \tag{8}$$

We say that the feedback controller (7) is γ-suboptimal if the closed-loop system (8) is stable and input-output stable with (2). Since the system (8) is of the form (1), we can apply Proposition 2. 1 to (8). The feedback controller (7) is γ-suboptimal if there exist common matrices $X_1 > 0$ and $X_2 > 0$ such that

$$\begin{aligned}&(A_i + B_{2i}F_j)^T X_1 + X_1(A_i + B_{2i}F_j) + X_2 + C_i^T C_i\\ &\quad + F_i^T F_i + X_1(\frac{1}{\gamma^2}B_{1i}B_{1i}^T + E_i X_2^{-1} E_i^T)X_1\\ &\quad = -P_{ij} < 0, \ i = 1, \cdots, r,\end{aligned}$$

for some $P_{ij} > 0$. But these conditions are rather strong and we relax them in the following theorem.

Theorem 2.1 *If there exist common matrices $X_1 > 0$ and $X_2 > 0$ such that*

$$\begin{aligned}
&(A_i + B_{2i}F_i)^T X_1 + X_1(A_i + B_{2i}F_i) + X_2 \\
&+X_1(\frac{1}{\gamma^2}B_{1i}B_{1i}^T + E_i X_2^{-1} E_i^T)X_1 \\
&+C_i^T C_i + F_i^T F_i = -P_i < 0,\ i = 1, \cdots, r, \\
&-\Delta F_{ij}^T \Delta B_{2ij}^T - \Delta B_{2ij}\Delta F_{ij} \\
&< P_i + P_j + \Delta C_{ij}^T \Delta C_{ij} + \Delta F_{ij}^T \Delta F_{ij} \\
&+X_1(\frac{1}{\gamma^2}\Delta B_{1ij}\Delta B_{1ij}^T + \Delta E_{ij} X_2^{-1} E_{ij}^T)X_1,\ i < j
\end{aligned} \tag{9}$$

for some $P_i > 0$, then the controller (7) is γ-suboptimal. Moreover, if further $B_{2i} = B_2$, $i = 1, \cdots, r$, then the second condition of (9) is always satisfied.

If we let $F_i = -B_{2i}^T X_1$, $i = 1, \cdots, r$ in (9), we obtain the following Riccati ineualities

$$\begin{aligned}
&A_i^T X_1 + X_1^T A_i + X_2 + C_i^T C_i \\
&+X_1^T(\frac{1}{\gamma^2}B_{1i}B_{1i}^T + E_i X_2^{-1} E_i^T - B_{2i}B_{2i}^T)X_1 \\
&= -P_i < 0,\ i = 1, \cdots, r.
\end{aligned} \tag{10}$$

As a design method, we have:

Corollary 2.1 *If there exist common matrices $X_1 > 0$ and $X_2 > 0$ such that (9) with its first condition replaced by (10) is satisfied, then the controller (7) with $F_i = -B_{2i}^T X_1$ is γ-suboptimal. Moreover, if further $B_{2i} = B_2$, $i = 1, \cdots, r$, then the second condition of (9) is always satisfied and the controller $u(t) = -B_2^T X_1 x(t)$ is γ-suboptimal.*

2.3 H_∞-Control with Output Feedback

Consider the system described by the fuzzy rules:

$$\begin{aligned}
\textbf{IF}\quad & \xi_1 \text{ is } M_{i1} \text{ and } \cdots \text{ and } \xi_q \text{ is } M_{iq}, \\
\textbf{THEN}\quad & \dot{x}(t) = A_i x(t) + E_i x(t-h) \\
& \quad +B_{1i}w(t) + B_{2i}u(t),\ i = 1, \cdots, r \\
& z(t) = \begin{bmatrix} C_{1i}x(t) \\ D_{12i}u(t) \end{bmatrix}, \\
& y(t) = C_{2i}x(t) + D_{21i}w(t)
\end{aligned} \tag{11}$$

where $u \in \mathcal{R}^{m_2}$ is the control input, $y \in \mathcal{R}^l$ is the observation. The matrices A_i, B_{1i}, B_{2i}, C_{1i}, C_{2i}, D_{12i}, D_{21i} and E_i are of appropriate dimensions. Here we assume that the premise variables do not depend on u. For simplicity we assume

$$D_{12i}^T D_{12i} = I,\ D_{21i}[B_{1i}^T\ \ D_{21i}^T] = [0\ \ I],\ \ i = 1, \cdots, r.$$

The state, controlled output and observation are defined as follows:

$$\begin{aligned}
\dot{x}(t) &= \sum_{i=1}^r \lambda_i(\xi(t))\{A_i x(t) + E_i x(t-h) \\
&\quad +B_{1i}w(t) + B_{2i}u(t)\},
\end{aligned} \tag{12}$$

$$z(t) = \sum_{i=1}^r \lambda_i(\xi(t)) \begin{bmatrix} C_{1i}x(t) \\ D_{12i}u(t) \end{bmatrix}, \tag{13}$$

$$y(t) = \sum_{i=1}^r \lambda_i(\xi(t))\{C_{2i}x(t) + D_{21i}w(t)\}.$$

Suppose that the following rules concerning H_∞ controllers for each subsystem (11) are given.

$$\begin{aligned}
\textbf{IF}\quad & \xi_1 \text{ is } M_{i1} \text{ and } \cdots \text{ and } \xi_q \text{ is } M_{iq}, \\
\textbf{THEN}\quad & \dot{\hat{x}}(t) = \hat{A}_{1i}\hat{x}(t) + \hat{A}_{2i}\hat{x}(t-h) + \hat{B}_i y(t), \\
& u(t) = \hat{C}_i \hat{x}(t),\ \ i = 1, \cdots, r
\end{aligned}$$

where $\hat{x} \in \mathbf{R}^{\hat{n}}$ and all matrices are of compatible dimensions. Then an actual choice of an H_∞ controller is

$$\begin{aligned}
\dot{\hat{x}}(t) &= \sum_{i=1}^r \lambda_i(\xi(t))\{\hat{A}_{1i}\hat{x}(t) + \hat{A}_{2i}\hat{x}(t-h) + \hat{B}_i y(t)\}, \\
u(t) &= \sum_{i=1}^r \lambda_i(\xi(t))\hat{C}_i\hat{x}(t).
\end{aligned} \tag{14}$$

Then the closed-loop system (13) with (14) can be written as

$$\begin{aligned}
\dot{x}_{cl}(t) &= \sum_{i=1}^r\sum_{j=1}^r \lambda_i(\xi(t))\lambda_j(\xi(t))(\mathcal{A}_{1cl}^{ij}x_{cl}(t) \\
&\qquad +\mathcal{A}_{2cl}^{ij}x_{cl}(t-h) + \mathcal{B}_{cl}^{ij}w(t)), \\
z(t) &= \sum_{i=1}^r\sum_{j=1}^r \lambda_i(\xi(t))\lambda_j(\xi(t))\mathcal{C}_{cl}^{ij}x_{cl}(t).
\end{aligned} \tag{15}$$

where $x_{cl}^T = \begin{bmatrix} x^T & \hat{x}^T \end{bmatrix}^T$ and

$$\mathcal{A}_{1cl}^{ij} = \begin{bmatrix} A_{1i} & B_{2i}\hat{C}_j \\ \hat{B}_i C_{2j} & \hat{A}_{1i} \end{bmatrix},\ \mathcal{A}_{2cl}^{ij} = \begin{bmatrix} E_i & 0 \\ 0 & \hat{A}_{2i} \end{bmatrix},$$
$$\mathcal{B}_{cl}^{ij} = \begin{bmatrix} B_{1i} \\ \hat{B}_i D_{21j} \end{bmatrix},\ \mathcal{C}_{cl}^{ij} = \begin{bmatrix} C_{1i} & 0 \\ 0 & D_{12i}\hat{C}_j \end{bmatrix}.$$

We say the feedback controller (14) is γ-suboptimal if the closed-loop system (15) is admissible and input-output stable with (2). For a given controller (14) we can apply Proposition 2.1 to (15) replacing A_i, E_i, B_i and C_i by $\mathcal{A}_{1cl}^{ij}, \mathcal{A}_{2cl}^{ij}, \mathcal{B}_{cl}^{ij}$ and $\mathcal{C}_{cl}^{ij}$, respectively. Now we propose a method to design a γ-suboptimal controller by applying the H_∞ theory to each subsystem.

Theorem 2.2 *Consider the system (13) with $D_{12i} = D_{12}$ and $D_{21i} = D_{21}$, $i = 1, 2, \cdots, r$. Suppose that there exist common matrices $X_1 > 0$, $X_2 > 0$, $Z_1 > 0$ and $Z_2 > 0$ such that*

$$\begin{aligned}
&A_i^T X_1 + X_1 A_i + X_2 + C_{1i}^T C_{1i} \\
&+X_1(\frac{1}{\gamma^2}B_{1i}B_{1i}^T + E_i X_2^{-1} E_i^T - B_{2i}B_{2i}^T)X_1 \\
&= -P_i < 0,\ i = 1, \cdots, r,
\end{aligned} \tag{16}$$

$$\begin{aligned}
&X_1\Delta B_{2ij}\Delta B_{2ij}^T X_1 - \{P_i + P_j + \Delta C_{1ij}^T \Delta C_{1ij} \\
&+X_1(\frac{1}{\gamma^2}\Delta B_{1ij}\Delta B_{1ij}^T + \Delta E_{ij} X_2^{-1}\Delta E_{ij}^T)X_1\} \\
&= -P_{ij} < 0,\ i < j
\end{aligned} \tag{17}$$

$$
\begin{aligned}
&(A_i + \frac{1}{\gamma^2} B_{1i} B_{1i}^T X_1) Z_1 + Z_1 (A_i + \frac{1}{\gamma^2} B_{1i} B_{1i}^T X_1)^T \\
&\quad + Z_1 Z_2^{-1} Z_1 + B_{1i} B_{1i}^T + E_i Z_2 E_i^T \\
&\quad + Z_1 (\frac{1}{\gamma^2} X_1 B_{2i} B_{2i}^T X_1 - C_{2i}^T C_{2i}) Z_1 \\
&\qquad\qquad = -Q_i < 0, \ i < j
\end{aligned} \tag{18}
$$

$$
\begin{aligned}
&Z_1 \Delta C_{2ij}^T \Delta C_{2ij} Z_1 - \frac{1}{\gamma^2} (Z_1 X_1 \Delta B_{1ij} \Delta B_{1ij}^T \\
&\quad + \Delta B_{1ij} \Delta B_{1ij}^T X_1 Z_1) - (Q_i + Q_j \\
&\quad + \Delta B_{1ij} \Delta B_{1ij}^T + \Delta E_{ij} Z_2 \Delta E_{ij}^T \\
&\quad + \frac{1}{\gamma^2} Z_1 X_1 \Delta B_{1ij} \Delta B_{1ij}^T Z_1 X_1) \\
&\qquad\qquad = -Q_{ij} < 0, \ i < j
\end{aligned} \tag{19}
$$

where $\Delta B_{1ij} = B_{1i} - B_{1j}$, $\Delta B_{2ij} = B_{2i} - B_{2j}$, $\Delta C_{1ij} = C_{1i} - C_{1j}$, $\Delta C_{2ij} = C_{2i} - C_{2j}$ *and* $P_i > 0$, $P_{ij} > 0$, $Q_i > 0$, $Q_{ij} > 0$. *Then the controller*

$$
\begin{aligned}
\dot{\hat{x}}(t) &= \sum_{i=1}^{r} \lambda_i(\xi(t)) \{A_i \hat{x}(t) + E_i x(t-h) \\
&\quad + B_{1i} \hat{w}(t) + B_{2i} u(t) + Z_1 C_{2i}^T (y(t) - \hat{y}(t))\}, \\
u(t) &= -\sum_{i=1}^{r} \lambda_i(\xi(t)) B_{2i}^T X \hat{x}(t)
\end{aligned} \tag{20}
$$

is γ*-suboptimal where*

$$
\begin{aligned}
\hat{y}(t) &= \sum_{i=1}^{r} \lambda_i(\xi(t)) C_{2i} \hat{x}(t), \\
\hat{w}(t) &= \frac{1}{\gamma^2} \sum_{i=1}^{r} \lambda_i(\xi(t)) B_{1i}^T X \hat{x}(t).
\end{aligned}
$$

Remark 2.1 *The conditions (16) and (18) are necessary and sufficient conditions such that a* γ*-suboptimal controller for the i-th subsystem in (11) exists. In this case, the* γ*-suboptimal controller is given by (20) with r being one.*

Next we consider a special case of the system (13).

Theorem 2.3 *Consider the system (13) with* $B_{2i} = B_2$ *and* $C_{2i} = C_2$, $i = 1, 2, \cdots, r$. *Suppose that there exist common matrices* $X > 0$ *and* $Z > 0$ *which satisfy (16) and (18). Then the controller*

$$
\begin{aligned}
\dot{\hat{x}}(t) &= \sum_{i=1}^{r} \lambda_i(\xi) [A_i + (\frac{1}{\gamma^2} B_{1i} B_{1i}^T - B_2 B_2^T) X_1 \\
&\quad - Z_1 C_2^T C_2] \hat{x}(t) + E_i \hat{x}(t-h) + Z_2 C_2^T y(t), \\
u(t) &= -B_2^T X_1 \hat{x}(t)
\end{aligned}
$$

is γ*-suboptimal.*

3 Conclusion

We have considered the H_∞-control problems for Takagi-Sugeno fuzzy time-delay models. First we have introduced a class of fuzzy time-delay models. Then we have given design methods for both H_∞-state and output feedback problems.

References

[1] T. Takagi and M. Sugeno (1985), Fuzzy identification of systems and its applications to modeling and control, IEEE Transaction on Systems, Man, Cybernetics 15, 116-132.

[2] K. Tanaka and M. Sugeno (1992), Stability analysis and design of fuzzy control systems, Fuzzy Sets and Systems 45, 135-156.

[3] K. Tanaka, T. Ikeda and H. O. Wang (1996), Robust stabilization of a class of uncertain nonlinear systems via fuzzy control: quadratic stabilizability, H^∞ control theory, and linear matrix inequalities, IEEE Transaction on Fuzzy Systems 4, 1-13.

[4] S.-K. Hong and R. Langari (1998), Synthesis of an LMI-based fuzzy control system with guaranteed optimal H_∞ performance, Proc. of Fuzz-IEEE '98, pp.422-427.

[5] G. Cao, W. Rees and G. Feng (1996), H_∞ control of nonlinear continuous-time systems based on dynamical fuzzy models, International Journal of Systems Science 27, 821-830.

[6] G. Feng, G. Cao and W. Rees (1996), An approach to H_∞ control of a class of nonlinear systems, Automatica 32, 1469-1474.

[7] J. Yoneyama, M. Nishikawa, H. Katayama and A. Ichikawa, H_∞-control for Takagi-Sugeno fuzzy systems, to appear in International Journal of Systems Science.

[8] M.C. Delfour (1984), Linear optimal control of systems with state and control variable delays, Automatica 20, 69-77.

[9] J.S. Gibson (1983), Linear-quadratic optimal control of hereditary differential systems: infinite dimensional Riccati equations and numerical approximations, SIAM Journal of Control and Optimization 21, 95-139.

[10] T. Kubo and E. Shimemura (1997), LQ regulator of systems with time-delays in states by memoryless feedback, SICE Transactions 33, 488-493(in Japanese).

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Function of General Regularization Term : Case Study on Two-Spiral Classification Problem

Weishui Wan Kotaro Hirasawa
Intelligent Control Laboratory, Graduate School of Information Science and Electrical Engineering, Kyushu University, 6-10-1, Hakozaki, Higashi-ku, Fukuoka, 812-8581, Japan
Email: hirasawa@ees.kyushu-u.ac.jp

Junichi Murata Jinglu Hu
Intelligent Control Laboratory, Graduate School of Information Science and Electrical Engineering, Kyushu University, 6-10-1, Hakozaki, Higashi-ku, Fukuoka, 812-8581, Japan
Email: murata@ees.kyushu-u.ac.jp

Abstract

What is the function of regularizers in the neural networks? What is the potential reason which produces the difference among different regularizers? Few papers exist on this aspect, and existent theoretical analyses are difficult to be used for guiding the application of regularizers to solve the real problems. In this paper we try to use different regularizers when training the neural networks to classify the typical two-spiral data bench as a case study to offer some clues to the above problem. Our simulation results suggest that (1) various regularizers tend to converge and generalize better than that without regularizers; (2) the difference of the convergence rate among different regularizers can be attributed to the complexity of regularizers, correspondingly the complexity of its derived ordered derivatives; (3) regularizers may be sensitive to the initialization of weight matrix of networks.

1 Introduction

It is well known by using an appropriate regularizer added to the error function in training the neural networks, one can always get a better generalization results than the one without the regularizer. After adding a regularizer, the citerion becomes

$$F^{new} = F^{original} + \alpha * \text{Regularizer term}$$

where α is a relative weight. Many regularizers have been proposed, such as the Laplace regularizer, a first-order term of network weights, and Gaussian regularizer, a second-order term of network weights, and other hybrid regularizers which combine the Gaussian and Laplace regularizers [1]etc. There exist already many sucessful applications of these regularizers to solve many difficult problems, but the reason explaining the different results caused by different regularizers remains unknown. Although there were some papers on this aspect[2][3], these comparisons are too simple to draw any conclusion on the difference of regularizers. In paper[1] the author proposed a hybrid method which incorporates the Laplace and Gaussian regularizers to find an optimal regularizer for a specific application based on the theoretical generalization error estimation, but this method needs a vast amount of computation before getting an optimal regularizer which makes it very difficult to be applied in the applications. In this paper a new different view is adopted to solve this problem. As a case study of regularizers in the training of neural networks, various regularizers are used when training the neural networks to classify the typical two-spiral data bench[5], this is a very difficult problem by just using the basic backpropagation algorithm. Our aim of simulations is to see , at least experimentally what is the latent reason which makes the difference among various regularizers. Our simulation results suggest that (1) various regularizers tend to converge and generalize better than that without regularizers; (2) the difference of the convergence rate among different regularizers can be attributed to the complexity of regularizers , correspondingly the complexity of its derived ordered derivatives; (3) regularizers may be sensitive to the initialization of weight matrix of networks.

This paper is organized as follows, in Section 2 the basic structure of neural networks and forms of regularizers which are used in the simulations are introduced, Section 3 is about the simulation results, observed phenomena and explanations. The final section of this paper is conlusions.

2 Basic structure and regularizer form

Regularizer, a term added to the error criterion function with the aim to reduce the complexity of neural networks in order to get better generalization results on the data which are not used as the training data, is now a common technique in the training of neural networks, especially when the basic backpropagation algorithm is used to train the neural networks. This is because the backpropagation tend to possess the lower pruning ability, so it has an inferior representation and generalization ability among all the known training algorithms. There are two basic regularizers. The first one is Laplace regularizer (form 1 regularizer in the following of this paper), that is the first-order term of network weights, and the second one is Gaussian regularizer(form2), that is the second-order term of network weights. These two regularizers are widely used sucessfully in real applications, and have profound theoretical support from Bayes theory[4].

In order to explore the function of regularizers during the training of neural networks, the following basic feed-forward neural network structure is adopted: 3 layers with node number: 2:25:20:1, hidden node function is sigmoidal, output function is linear function. The following forms of regularizers are used in simulations:

$$form1 = \sum_{ij} \mid w_{ij} \mid, \quad (1)$$

$$form2 = \sum_{ij} w_{ij}^2, \quad (2)$$

$$form3 = \sum_{ij} \frac{w_{ij}^2}{1 + w_{ij}^2}, \quad (3)$$

$$form4 = \sum_{ij} \exp\{\beta \times w_{ij}^2\} \quad (4)$$

where i, j is the index of nodes in neural networks, w_{ij} is the connection weights, β is the predefined parameter. The training algorithm is Backpropagation algorithm, although other efficient algorithms can be used in the simulations. The task to be solved is the famous two-spiral data bench[5]. The reasons for using the backpropagation algorithm are that the two-spiral data bench is well known to be very difficult to solve by backpropagation, while this data bench is easy to solve by using the variants of backpropagation such as Levenberg-Marquardt backpropagation[4]. A simple introduction of the above regularizers is described here. Regularizer form 1 is the Laplace regularizer, regularizer form 2 is the Gaussian regularizer, regularizer form 3 is the weight elimination regularizer, and regularizer form 4 is a new regularizer used in this paper to explore the effect of the regularizer. It is evident the complexity of derivatives of the above regularizer's forms is in order such as form 1 → form 2→ form 3→ form 4 from small to large . We will find there is a close relation between the complexity of regularizer and convergence time needed when using the above regularizers in training the neural networks in the next section, although there is no evident difference among the effect on the performance.

3 Simulation results

The two-spiral problem is a very difficult problem to solve with the basic Backpropagation in the reasonable time[5], even with much more powerful Radial Basis Function. A lot of papers have been published for solving this typical problem. The final page of this paper is about the simulation results obtained by using the above 4 kinds of regularizers when training the neural networks with the backpropagation algorithm. The figures on the left side are the learning curves of regularizers form 1, form 2, form 3, form 3 and form 4 from upper to bottom respectively, while the right side is about the corresponding generalization results. Note that two run results are listed for the regularizer form 3, this is because regularizer form 3 behaves instably. The other parameters used in simulations are set as follows: $\alpha = 0.0031, \beta = 0.004$. The typical times of one epoch by using the above 4 regularizers are as follows(in seconds):

regularizer	time of one epoch
form 1	10.6
form 2	14.1
form 3	15.7
form 4	18.3

From simulations(Figures 1,3,5,7,9) and the above table we can see that a great difference exists among the total time needed to finally get a good solution,which sucessfully classifies the two-spiral data sample in simulations, and the total time obtained by multiplying the number of iteration steps by the time spent in each epoch of the algorithm is in order such as : form 1 → form 2→ form 3→ form 4 from small to large in the above simulation environment. Aslo one can observe that after some number of iteration steps, two-spiral problem can be always solved in the meaning that not only the original data is completely represented, but also the final neural network generalizes well in the new points for the regularizers form 1, form 2 and form 4. While for the regularizer form 3(Figures 5,6,7,8 on final page), a good solution

is not always obtained, sometimes two-spiral problem is solved sucessfully, sometimes not. The reason can be partly explained by the time spent for computing the derivatives of the criterion function(error function +regular term) with respect to connection weights using the backpropagation algorithm. The derivatives of the above regularizers with respect to a specific weight w_{ij} are listed respectively as follows:

$$\frac{\partial form1}{\partial w_{ij}} = \begin{cases} 1, & \text{if } w_{ij} > 0 \\ -1, & \text{if } w_{ij} < 0 \\ 0, & \text{if } w_{ij} = 0 \end{cases}, \quad (5)$$

$$\frac{\partial form2}{\partial w_{ij}} = 2 * w_{ij}, \quad (6)$$

$$\frac{\partial form3}{\partial w_{ij}} = \frac{2 * w_{ij}}{(1 + w_{ij}^2)^2}, \quad (7)$$

$$\frac{\partial form4}{\partial w_{ij}} = \exp\{\beta \times w_{ij}^2\} * 2 * \beta * w_{ij} \quad (8)$$

Evidently the complexity of the derivatives of the above regularization forms is in order such as form 1 $\rightarrow$ form 2$\rightarrow$ form 3$\rightarrow$ form 4 from small to large. The relation between the complexity of the partial derivatives of regularizers with respect to network weights and the convergence time needed to sucessfully classify the two-spiral problem can be partly explained as follows. From the weight iteration formula of the backpropagatiton, it is easily seen the main part of the algorithm lies in the computation of derivatives. For the output layer, this computation is easy, while for the hidden layer, the derivatives are computed in the backward direction iteratively through the output layer to hidden layer. In this step ordered derivatives are used. The computation of derivatives for the nodes in the output layer and hidden layers is the core of backpropagation algorithm. The time spent in this part directly determines the whole convergence time consumed to get a good solution. Therefore a small time needed to compute the derivatives means a short time to convergence to the true solution under the same network structure and same computing environment. The reason for the instability when using the regularizer form 3 remains to be explored. The instability of regularizer form 3(from the figures 5,6,7,8 on the final page) says that this kind of regularizer is sensitive to the initialization of network weights. From the simulations, the learning curves of two experiments are similar, but the generalization results of these two experiments are different. This can be explained by the difference between the representation capability of networks and generalization capability of networks. These two concepts are closely related, but not the same, a good representation capability of the networks does not always mean a good generalization capability. In other words the instability of regularizer form 3 says that the final structures of the networks obtained after training are different, so their generalization on the new data points are different. A careful selection of initialization of weight matrix maybe needed in order to guarantee the convergence of the algorithm.

4 Conclusions

In this paper we try to use various regularizers when training the neural network to classify the typical two-spiral data bench[5] as a case study to offer some clues to answer the above problem proposed at the beginning of this paper, i.e. what is the function of various regularizers in the training of neural networks and what is the latent reasons which produce the difference among various regularizers reported when applying these regularizers to real problems. Simulation results of this case study suggest that there is probably no necessicity to find the optimal regularizer as done recently by Professor Ishikawa[1]. One can solve this problem through the selection of the appropriate parameters , such as learning rate etc. in the backpropagation algorithm , although the selection of parameters in the learning algorithms is very difficult. Fortunately for the backpropagation algorithm there exist many empirical rules which can be used. Further simulations are being done and theoretical explanation for all the phenomena observed in the simulations is being explored.

References

[1] M.Ishikawa, H.Shimada and S.Amari, "Iterative design of regularizers based on data by minimizing generalization errors", *Proceedings of IEEE-INNS-ENNS international joint conference on neural network.*, vol 1, pp.3-9, 2000, Como, Italy

[2] C.Goutte and L.K.Hansen, "Regularization with a pruning prior", *Neural networks*, Vol.10, No.6, pp.1053-1059(1997)

[3] K.Hensen and C.E.Rasmussen, "Pruning from adaptive regularization", *Neural Computation*, Vol.6, pp.1223-1232 (1994)

[4] MacKay, Neural Computation, vol. 4, no. 3, 1992, pp. 415-447

[5] K.Lang and M.Witbrock,"Learning to tell two spirals apart", in *Proceedings of the 1988 Connectionist Models Summer School*, Morgan Kaufmann, 1989.

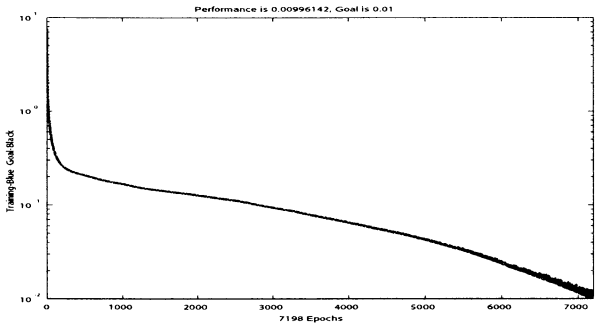


Figure 1: Leaning Curve of form 1

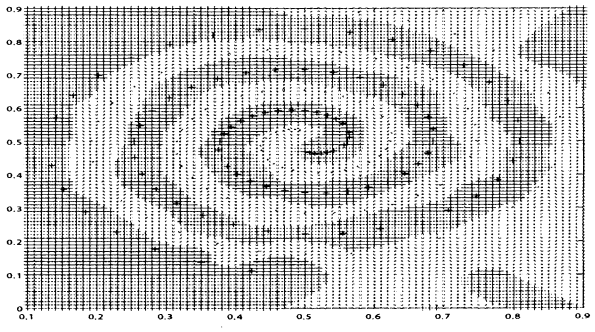


Figure 2: Generlaization Results of form 1

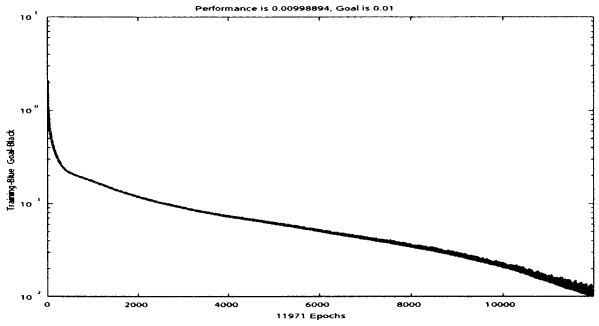


Figure 3: Leaning Curve of form 2

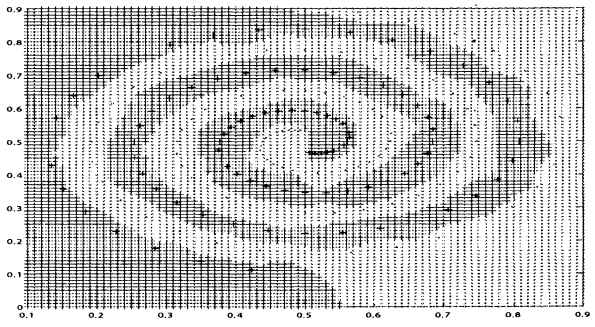


Figure 4: Generlaization Results of form 2

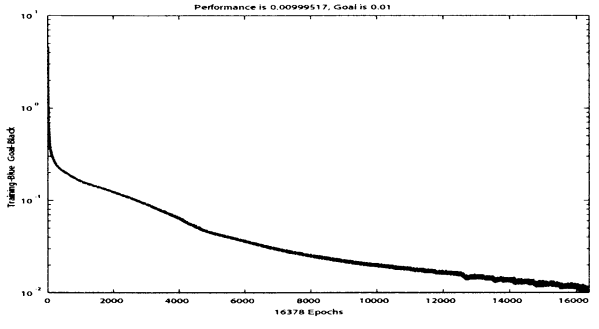


Figure 5: Leaning Curve of form 3(case1)

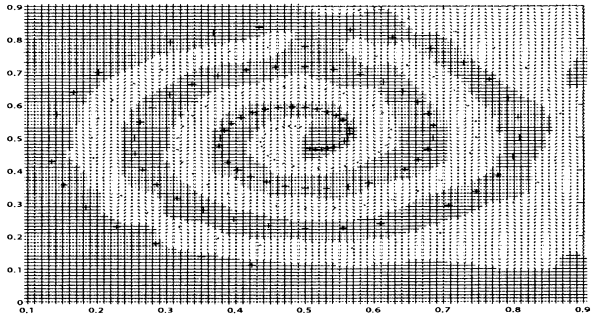


Figure 6: Generlaization Results of form 3(case1)

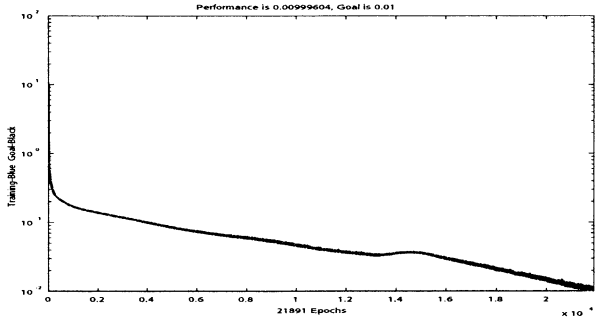


Figure 7: Leaning Curve of form 3(case2)

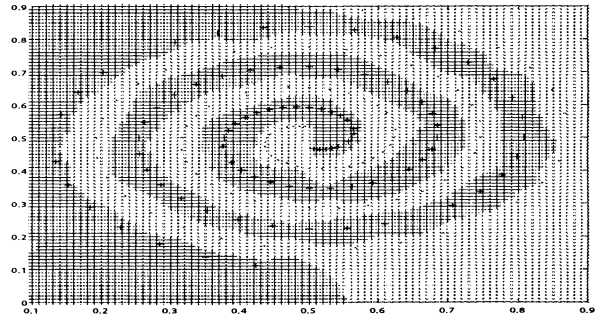


Figure 8: Generlaization Results of form 3(case2)

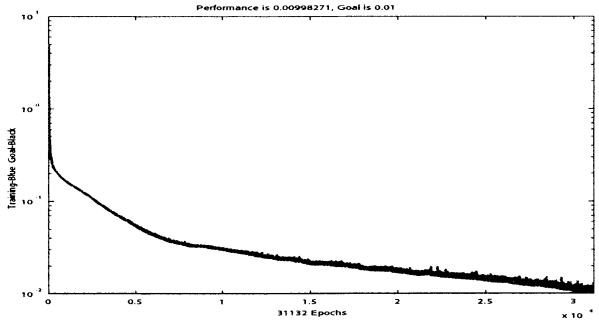


Figure 9: Leaning Curve of form 4

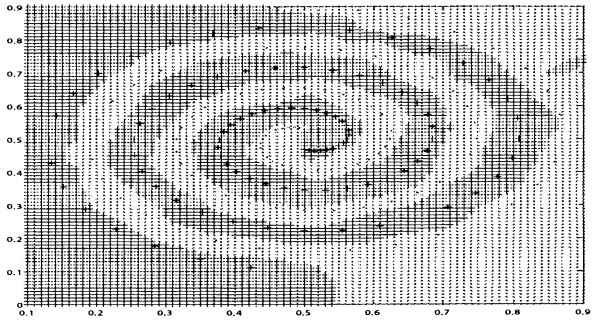


Figure 10: Generlaization Results of form 4

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Function Approximation Using LVQ and Fuzzy Set

Shon Min-Kyu, Junichi Murata, Kotaro Hirasawa
Department of Electrical and Electronic Systems Engineering
Intelligent Control Laboratory
Kyushu University,6-10-1,Hakozaki,Higashi-ku,Fukuoka,812-8581,Japan

Abstract

Neural networks with local activation functions, for example LVQ(Learning Vector Quantization) and RBFN(Radial Basis Function Network), have a merit of excellent generalization abilities. When this type of network is used in function approximation, it is very important to determine the proper division of the input space into local regions to each of which a local activation function is assigned. In RBFN, this is equivalent to determination of the locations and the numbers of its RBFs, which is generally done based on the distribution of input data. But, in function approximation, the output information (the value of the function to be approximated) must be considered in determination of the local regions. A new method is proposed that uses LVQ network to approximate the functions. It divides the input space into regions with a prototype vector at the center of each region. The ordinary LVQ, however, outputs discrete values only, and therefore can not approximate continuous functions. In this paper, fuzzy sets are employed in both of learning and output calculation. A few examples are provided to show the effectiveness of the proposed method.

1 introduction

Neural networks with local activation functions, for example RBFN (Radial Basis Function Network), have a merit of excellent generalization abilities. So they have been widely used in function approximation.When this type of network is used in function approximation, it is very important to determine the proper division of the input space into local regions to each of which a local activation function is assigned. In RBFN, this is equivalent to determination of the locations and the numbers of its RBFs, which is generally done based on the distribution of input data by using, for example, clustering technique; more number of RBFs are placed where the input data are dense while the areas where the input data are sparsely distributed have fewer RBFs. Then the network weights are adjusted to minimize the approximation errors. So, the local regions are determined by the input information, and the output information is used only for the later adjustment. However, in function approximation, the above procedure does not work well. Consider a set of input data which are uniformly distributed. Then, the above procedure will give a uniformly divided input space. But, in the areas where the value of the function to be approximated changes violently, there should be a lot of small local regions (a lot of RBFs) to obtain good approximation, while in the areas where the change of the function values is very small, fewer regions (fewer RBFs) will suffice. So, the output information (the value of the function to be approximated) must be considered in determination of the local regions.

Here, a new method is proposed that uses LVQ (Learning Vector Quantization) network to approximate the functions. LVQ also has a characteristic of local activation functions. The ordinary LVQ, however, outputs discrete values only, and therefore can not approximate real-valued functions. In this paper, fuzzy sets are employed in both of learning and output calculation. In the learning phase, unlike the ordinary LVQ, the true output value may not completely match any of the prototype labels because the true output can take any real value while there are only a finite number of prototypes. This possible error between the true value and the winner's label introduces fuzziness in the judgment of whether the winner is correct or not. A fuzzy set is assigned to each node to cope with this fuzziness. To calculate the network output for the input points in between the prototypes, interpolation is necessary and is done using another fuzzy set.

A few examples are provided to show the effectiveness of the proposed method.

2 LVQ Networks

2.1 LVQ Networks

Kohonen's LVQ network is a supervised learning algorithm associated with the competitive network shown in Fig.1. The network consists of an input layer and an output layer. A weight vector is associated with each node in the output layer as shown in Fig.1. The LVQ network calculates distance d between the

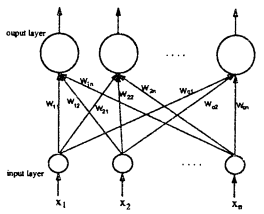


Fig.1. Structure of LVQ network

input vector $\boldsymbol{x}=(x_1,x_2,\cdots,x_p)$ and weight vector $\boldsymbol{w}_i$, $(i=1,\cdots,c)$ as in Eq(1)

$$d_j = \sqrt{\sum_{i=1}^{n}(x_i - w_{ji})^2}. \qquad (1)$$

Then, the LVQ network selects a weight vector $\boldsymbol{w}_k$, which minimizes distance d, and the selected weight vector $\boldsymbol{w}_k$ is trained as follows,

$$\begin{aligned} \boldsymbol{w}_k &:= \boldsymbol{w}_k + \Delta\boldsymbol{w}_k, \\ \Delta\boldsymbol{w}_k &= \pm\eta(\boldsymbol{x} - \boldsymbol{w}_k). \end{aligned} \qquad (2)$$

where $\eta > 0$ is the learning rate. In Eq.(2), if the selected output node k(the 'winner') is correct, the plus sign is adopted, and otherwise, the minus sign is used. After the learning, the LVQ network chooses the nearest weight vector to a given input vector, and outputs the 'label' corresponding to the weight vector as the network output. Thus, a weight vector can be regarded as the center of a local region in the input space.

In function approximation problems, a certain fixed real value must be assigned to each weight vector as its 'label'. So, if there are a sufficient number of weight vectors with various 'label' values, they found their proper locations by their learning. This realizes the division of the input space based on the output values of the training data.

2.2 Difficulties with LVQ algorithm in function approximation

However, when the ordinary LVQ algorithm is used in a function approximation, it has some problems. The LVQ training algorithm drives a weight vector closer to a training input vector when the 'label' value of the weight vector matches the training output value. But, there are only a finite number of 'labels' while the training output data can take any real number. So, there are an infinite number of data that do not match any weight vector 'label', and thus the LVQ learning algorithm in its original form does not work well in function approximation problems. Another problem occurs when the output is calculated after learning. The LVQ network output can take a value among the finite set of 'label' values only. So, the approximation of the training data shown in Fig.2 by the LVQ network is at best a piecewise step function as shown in Fig.3.

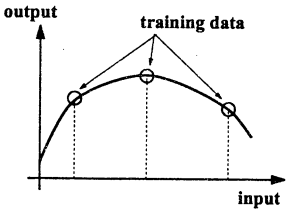


Fig.2. Function and Training data.

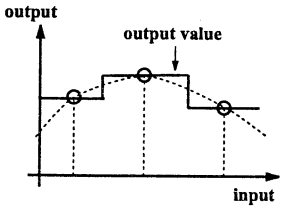


Fig.3. Difficulties with LVQ algorithm in function approximation.

3 Fusion of LVQ network and Fuzzy Sets

3.1 Fuzzy algorithm for Learning

The problem of the naive use of LVQ in function approximation arises from the gap between the discrete 'label' values assigned to the weight vectors and the continuous function values to be approximated.

In the learning phase, to overcome the problem, a fuzzy 'label' is assigned to each weight vector. Figure 4 shows the typical membership functions for the fuzzy 'labels'. In the figure, on the interval of possible output value [MIN, MAX], five fuzzy 'labels' are defined, where MIN, $n1$, $n2$, $n3$ and MAX are representative values of the fuzzy ' labels'. The LVQ

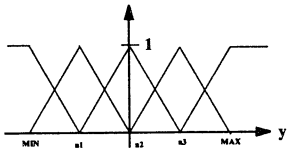


Fig.4. Membership functions.

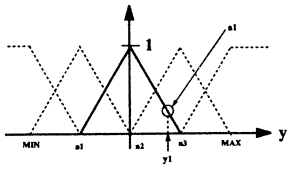


Fig.5. Use of membership function in learning.

learning described in Eq.(2) is modified so that the weight update is done based on the degree of matching of the 'label' and the training output value which is measured by the membership function. Figure 5 illustrates the idea. Suppose that the training output value is $y1$ and that the weight vector with the 'label' $n2$ is nearest to the training input. Then the degree of matching is calculated as $a1$ by the membership function. This degree a1 determines how close the weight vector can approach the current training input vector. The details are expressed by Eq.(3), (4) and (5),

$$\begin{aligned} \Delta \boldsymbol{w}_{ij} &= +(\eta(\boldsymbol{x}_i - \boldsymbol{w}_{ij}) \\ &\quad -sgn(\boldsymbol{x}_i - \boldsymbol{w}_{ij}) * M(a1)), \quad (3) \\ M(a1) &= E(1 - a1), \quad (4) \\ E &= (MAX - MIN)/n. \quad (5) \end{aligned}$$

When the weight vector converges, Δw_{ij} becomes zero. By equating the right hand side of Eq.(3) zero, we know that the distance between the weight w_{ij} and the input vector x_i can not be smaller than $M(a1)/\eta$. This is the limit on how close the weight vector can approach the input vector, and the limit depends on the degree of matching $a1$.

3.2 Fuzzy algorithm for output

The ordinary LVQ network outputs discrete values only, and therefore can not approximate a continuous function(Fig.3). In this paper, using fuzzy algorithm for output layer, the LVQ network can output continuous values and therefore approximate continuous functions.

For a give input vector, the LVQ network selects $p+1$ nearest weights to the input, and smooths their representative 'label' values using fuzzy sets defined on the input space.

The equation of fuzzy algorithm is shown by Eq.(6),

$$y = \frac{\sum_{i=1}^{p+1} (\frac{1}{||\boldsymbol{w}_i - \boldsymbol{x}||} * n_i)}{\sum_{i=1}^{p+1} \frac{1}{||\boldsymbol{w}_i - \boldsymbol{x}||}}, \quad (6)$$

where n_i is the representative value of 'label' of node i, and the dimension of input x is p. The procedure is illustrated by Fig.6. The fuzzy algorithm outputs the curve shown in Fig.7 which is in contrast to Fig.3 obtained by the ordinary LVQ network.

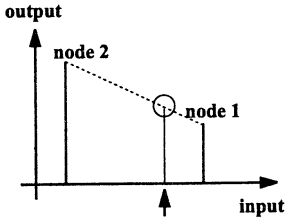


Fig.6. Output value among nodes.

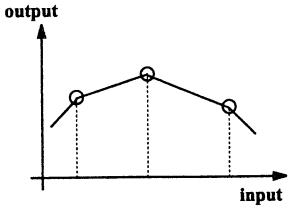


Fig.7. Output value using fuzzy algorithm.

4 Examples

In this paper, the following functions are approximated

$$z = x + y, \quad (7)$$

$$z = sin(2x) + cos(2y). \quad (8)$$

Fig.8 and Fig.10 show the traing data from the two functions. Each of examples are used 169 training data. The representative values are generated by $(MAX + MIN)/20$ and are assigned to the output nodes of LVQ network. The LVQ network consists of 600 output nodes. Results of function approximation using the proposed LVQ algorithm are shown in Fin.9 and Fig.11, which approximate the target function well. As the fuzzy membership function for the output layer are a stright lines, the approximated function is not smooth. Using the fuzzy sets of various shape can approximate a lot of function smoothly.

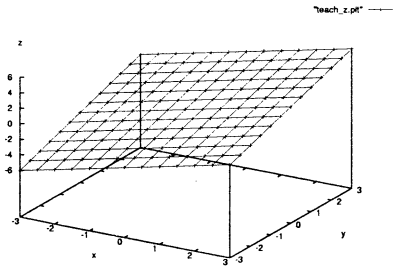


Fig.8. Training data produced by Eq.5.

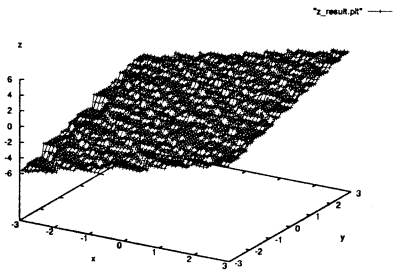


Fig.9. Result of learning of data in Fig.9

5 Discussions and Conclusions

This paper presents a method for approximation of functions using LVQ algorithm and fuzzy sets. More specifically, this paper investigates input space division by using the output infotmation and input information of training data. The effectiveness of the proposed method is verified by examples. In this method, one weight vector is updated for a given training data vector. But one data vector may contain useful information for updating more than one weight. If all of the weight vectors which correspond to a training data

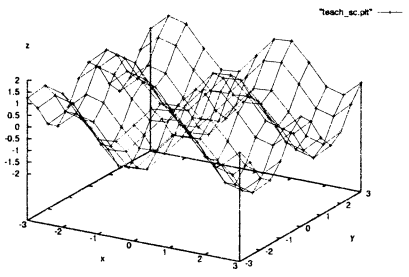


Fig.10.Training data produced by Eq.6.

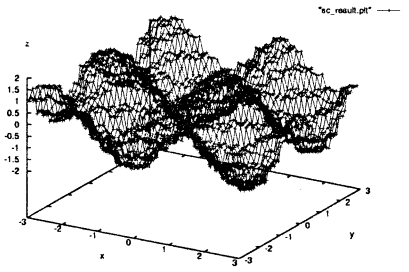


Fig.11. Result of learning of data in Fig.12.

vector are trained, more efficient function approximation will be realized.

References

[1] Kohonen,T, "Self-Organizing Maps, Springer Series in Information" *Sciences*, 1995.

[2] Kohonen,T, "Learning Vector Quantization for Pattern Recongition" *Technical Report TKK-F-A602, Helsinki University of Technology, Finland*, 1990.

[3] S.Chen, C.F.N.Cowan, and P.M.Grant. "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks" *IEEE Trans. Neural Network*,vol2,302-309, 1991.

[4] R. Katayama, Y.Kajitani, K.Kuwata, Y.Nishida, "Self generation radial basis function as neuro-fuzzy model and its application to nonlinear prediction of chaostic time series" *2nd Int. Conf on Fuzzy Systems (FUZZY-IEEE'93)*,407-414, 1993.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Robot Path Planning for Visiting FA Working Points by Obstacle Avoiding using GA

Hidehiko Yamamoto*and Daniel Moldovan**

* 1-1, Yanagido, Gifu-shi, Dept. of Mechanical and Systems Engineering, Faculty of Engineering, Gifu University, E-mail: yam-h@cc.gifu-u.ac.jp.

** 930 Sakaedani, Wakayama-shi, Dept. of Intelligent Systems, Faculty of Systems Engineering, Wakayama University, E-mail: c12s002@center.wakayama-u.ac.jp.

Abstract: The path-planning problem is of major interest for robotics. This paper describes a method of avoiding obstacles and finding the shortest path for a robot that travels between machine tools and obstacles in a production line FA factory. Evolving the GA over a population of non-lethal individuals will generate the robot path. A lethal individual represents a robot path that generates a collision between the robot and one of the obstacles or the machines. The algorithm uses an original technique for generating the populations of individuals. The method is applied to a flexible manufacturing system (FMS) with a variable number of machines and obstacles.

Keywords: path planning, obstacle avoiding, FA, GA

1. Introduction

A basic version of path-planning problem consists of finding a sequence of steps for a robot from a start point to a given point while avoiding collisions with any obstacles in the environment.
In the next generation production systems, the machine tools and robots will have different amounts and levels of knowledge and mobility that will be used for interaction between each other. The production plan will be achieved by changing their production configuration (using gathering or dispersing). Because of this changeable situation it is useless to program the robot path while developing the production system.
For any new factory configuration, the robot has to be able to find by itself the way to visit the machines in a predefined order and to avoid obstacles. It has to receive only the information about the machines and obstacles coordinates. After that, everything will be up to the program that will implement an algorithm in order to find the shortest robot path.

This paper addresses the problem of getting robots to learn to program themselves by means of genetic algorithm (GA), in order to find the shortest way to visit the machine working points in a predefined order and with avoiding the obstacles. The possible paths will be searched from a network of intermediate points. To accomplish this, the genetic algorithm starts with a population of randomly created robot paths. Then, the population is genetically breed using the Darwinian principle of survival and reproduction of the fittest and the genetic operation of sexual recombination (crossover) and mutation. The length of the robot path between two working points represents the selection criterion. The shorter the distance the better fitted is the individual. The genetic operators of crossover and mutation will be applied successively over randomly selected individuals. Over a period of many generations, we breed populations of paths that are ever more fit in solving the problem.

The path that will emerge from this genetic programming will be a consequence of fitness and will be highly fit in solving the planning problem.

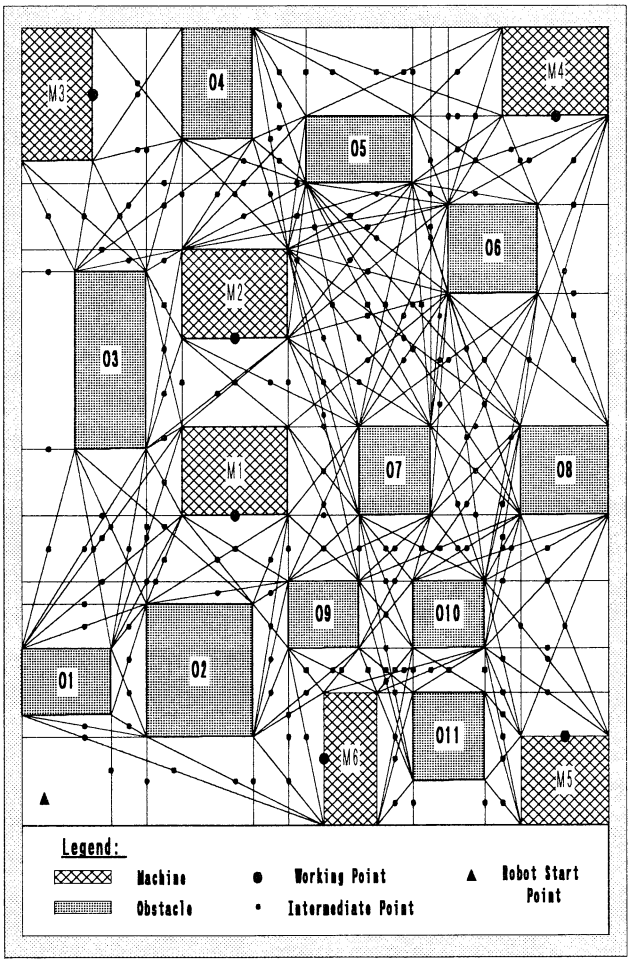


Fig. 1 The network of intermediate points

2. The network of intermediate points

In our approach, the factory is seen like a Cartesian space in which the machines and the obstacles have rectangular shapes. In order to find the path from one machine to another, the searching algorithm will process a global network of intermediate points. The points that are situated at the middle of the imaginary segments that connects the corners of the rectangles, localized face to face, represents the junctions of the network of intermediate points (see Fig. 1).

To this network we add the points situated at the middle of the segments drawn perpendicularly between the rectangle's corners and the sides of the factory (see Fig. 1) with the condition that those segments don't intersect any rectangle. If eventually, two intermediate points have the same coordinates, one will be dropped.

The junctions of this network are numbered and the sides of the network will represent the potential routes that can be followed by the robot in order to visit the machine working points localized on one side of the rectangles.

From this global network the genetic algorithm will choose the intermediate points that will be passed through by the robot in order to visit each machine and to avoid any obstacle.

3. The robot path

Our robot path representation scheme in this GA based processing is a list that contains the network joints that have to be traveled by the robot in order to visit two consecutive working points (see Fig. 2).

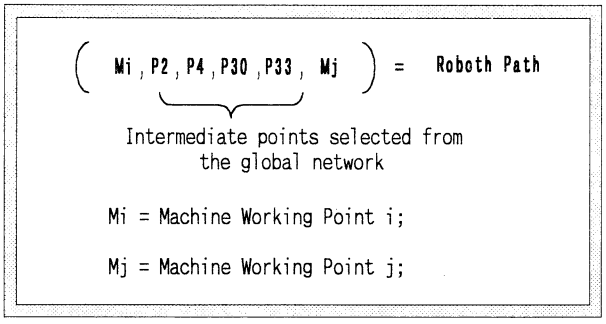


Fig. 2 The robot path

This list represents the basic element of our GA's population – the individual.

A distinct feature of our implementation is that only the insertion of the non-lethal individuals will compose the populations. The robot path generated by these individuals doesn't generate collisions either with the machines or with the obstacles.

For determining if an individual is lethal or not we will check to see if by unifying all of the network joints within the individual, the path obtained is not crossing any rectangles. This is done by verifying every intermediate segment within the robot path with every segment of every rectangle from the factory. This testing is a reiteration of determining if two segments are crossing each other. This can be solved by means of line equation as shown in Fig. 3.

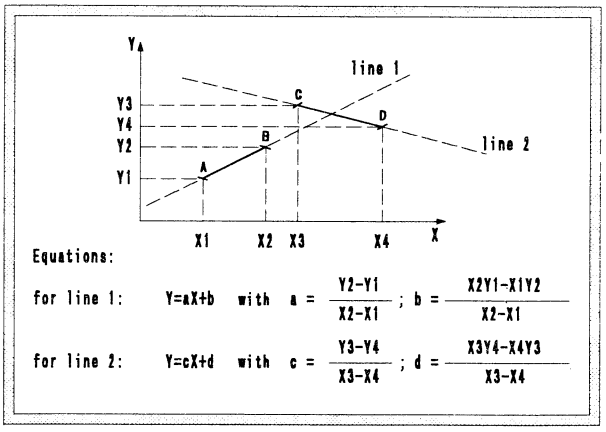


Fig. 3 Lines equations

The flowing chart of determining if two segments are crossing each other is presented in Fig. 4. We are using a function sgn(y,x) which determines the sign of the y-x expression. At the beginning we are establishing the position of the CD segment's ends towards line 1. This comes down to determining the signs of expressions E3=Y3-(aX3+b), corresponding to point C, and E4=Y4-(aX4+b) corresponding to point D. If these two expressions have the same sign, and therefore M1>0, means that points C and D are both on the same side of line 1. This implies that points C and D are also on the same side of the segment AB. This means that there isn't any intersection between segments AB and CD.

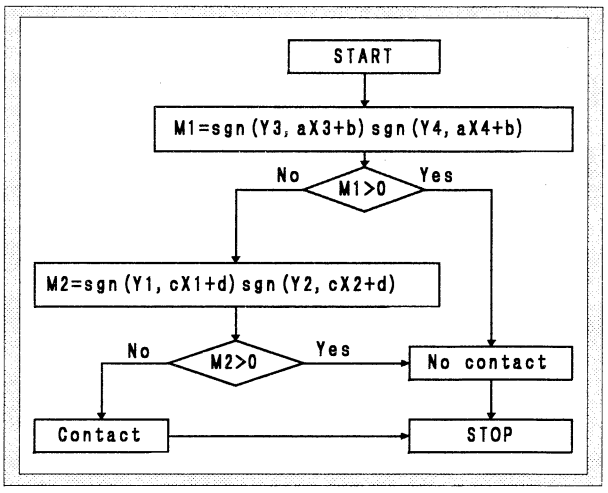


Fig. 4 Determining if two segments are crossing each other

If E3 and E4 have different signs, and therefore M1<=0, means that there is a contact point between CD and line 1. But this doesn't imply that CD has the

same position toward AB. Therefore we can't yet say that the segments are crossing or touching each other. The next step is represented by establishing the position of the AB segment's ends towards line 2. This comes down to determining the signs of the expressions E1=Y1-(cX1+d), corresponding to point A, and E2=Y2-(cX2+d), corresponding to point B. If E1 and E2 have the same sign, and therefore M2>0, means that points A and B are both on the same side of line 2. This implies that points A and B are also on the same side of the segment CD. This means that there isn't any intersection between AB and CD

On the other hand, if E1 and E2 have different signs, and therefore M2<=0, means that there is a contact point between segment AB and line 2. But because the segment CD has already a contact point with line 1 it comes out that the segments AB and CD have a contact point and so there is an intersection.

The length of intermediate segments not crossing any rectangle is then calculated. This length will be used in the end for calculating the individual fitness (the sum of all intermediate lengths).

4. The genetic operators

The primary operators involved in our genetic algorithm are: selection, crossover and mutation. Our selection criterion is the length of the robot path between two working points. The shorter the distance, the better fitted is the individual. The crossover operator creates new individuals by taking sub-lists from two randomly selected individuals to form an offspring. The crossover point is selected randomly each time the crossover operator is applied.

The, mutation operator creates new individuals by randomly altering one of the intermediate points within the individual selected randomly from the previous population. The mutation point is selected randomly each time the mutation operator is applied. The genetic algorithm will perform iteratively those operations on each generation of individuals to produce new generations of individuals, until a termination criterion is satisfied.

5. Obstacles avoiding with genetic algorithm

The primary parameters for controlling the genetic algorithm are the population size (M) and the maximum number of generations to be run (G). We are starting the genetic algorithm by creating an initial population of nonlethal individuals.

The flowchart of the process of creating the initial population is shown in Fig 5. We create the first non-lethal individual from this population by selecting, as intermediate points, the network joints that united will generate a robot path that will not intersect any of the rectangles. We continue with a cyclic process in which the individuals are generated by the computer. The intermediate points of these individuals will be selected randomly from the joints of the global network of intermediate points. Then, the resulted individuals are tested in order to detect their lethal properties. The lethal individuals will be abandoned and those that are non-lethal will be inserted in the initial population. This process is ended when the initial population reaches her predefined size (M).

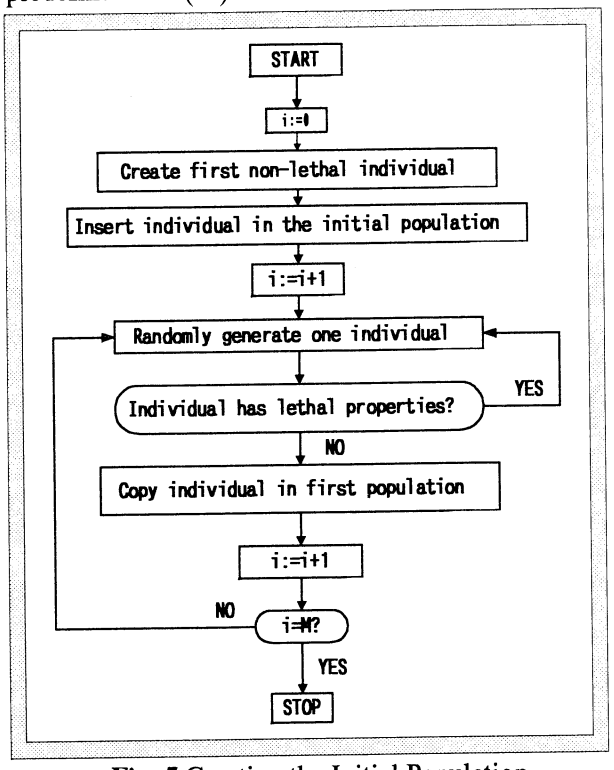


Fig. 5 Creating the Initial Population

After completing the process of creating the initial population, we continue the genetic algorithm by evaluating the fitness of each individual from that population. The best-fit individual will be selected as the first individual that will be inserted into the new population. The process of generating the new population is started by copying the best-fit individual from the previous population into the new population. Than we continue with the process of applying the crossover operator over randomly selected 2 individuals from the previous population. The resulting offsprings will be tested against lethal properties and only those, which have been passed that test, will be inserted into the new population. The other ones will be abandoned.

If at this moment, the population reaches her predefined size, the process of generating the new population will stop. Otherwise, we continue with applying the mutation operator over randomly selected individual from the previous population. The resulting offspring will be tested against lethal

properties and if it passes that test, it will be inserted into the new population. Otherwise it will be abandoned. At this moment, the population size will be tested again. If the new population reaches her predefined size, the process of generating the new population will stop. Otherwise, it will be reloaded from the intermediate step of applying the crossover operator.

After completing the process of generating the new population we continue with the process of testing the termination criterion. This termination criterion can be represented by the obtaining of a good value for fitness or by the reach of a certain number of generations. If the termination criterion is not satisfied we reload the algorithm from the intermediate step of generating of a new population. This process is then repeated till the termination criterion is satisfied. The algorithm is ending by designating the result– the best fitted individual.

6. Application example

We have tested the algorithm over the factory area presented in Fig. 1. The robot route has been divided into seven intermediate paths. The first six paths are corresponding to each visited working point and the seventh path is corresponding to returning to the starting point (see Fig. 6). We applied the algorithm considering next robot route: Start Point – M4 – M6 – M3 – M5 – M2 – M1 – Start Point.

The genetic algorithm has been applied on each intermediate path. We tested the algorithm for two different sizes of populations (see Fig. 7).

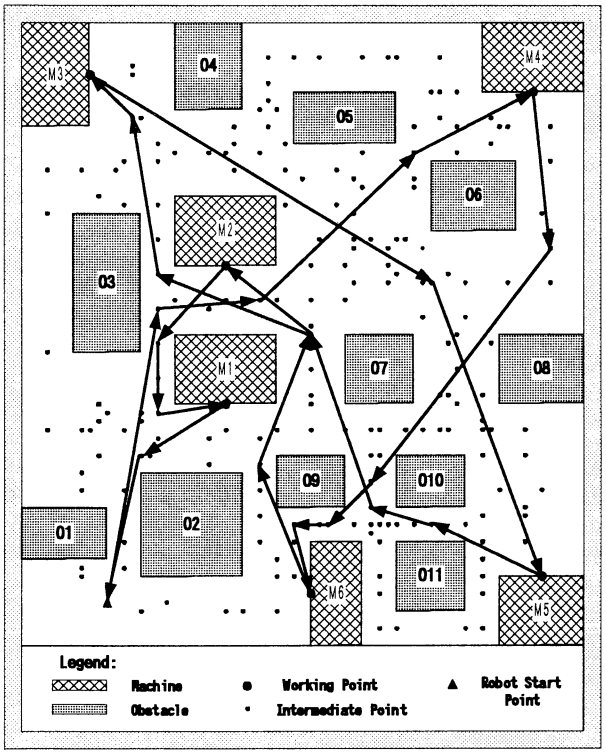


Fig. 6 The robot route

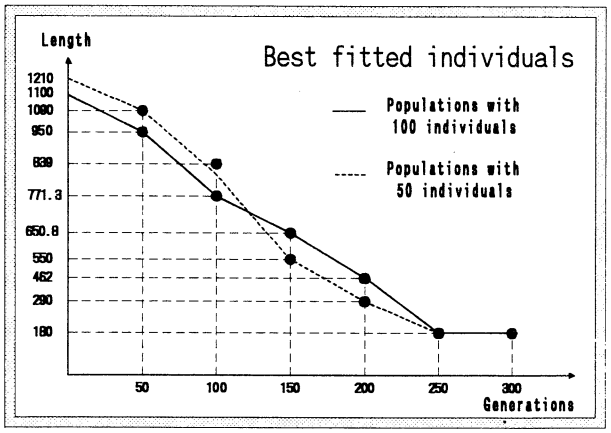


Fig. 7 Best fitted individuals

Each individual had a variable number of maximum eight and minimum one intermediate points. During the testing we noticed that from the 250th generation, the final result for all intermediate paths is already steady.

7. Conclusions

This paper described a method for generating automatically a robot path in order to avoid obstacles in a production line factory by genetic programming. In this method of programming, the successive genetic populations are made by choosing the best fit individual from the previous generation, followed by successive crossovers and mutations operated over the previous generation till the new population number reach its predefined value.

The method was tested with several simulation runs and the downward evolution of the best-fitted individuals length confirmed that there is an increased chance to find a shorter robot path once the generation number increases. The advantage of this solution is its simplicity in setting up an autonomous robot, capable of adapting and improving its operations in dynamic production systems. A limitation of the method is that once the number of intermediate points gets bigger - in order to have a smoother robot path - the time that is needed for finding the shortest path is getting higher.

Future works concentrates on fastening the method for determining if an individual is lethal or not and on implementing of a new method for improving the trajectory of the robot in order to avoid the zigzags and to make it smoother.

Reference

[1] John R. Koza and James P. Rice, Genetic Programming, The MIT Press, Cambridge

[2] Yamamoto H (1998) - Robot path planning by genetic programming. Artif Life Robotics (1998) 2:28-32.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Self-organizing map neural network as a multiple model identifier for time-varying plants

Alireza Fatehi[o], *Kenichi Abe*
Abe lab., Department of Electrical Engineering, Tohoku University
Aoba 05, Aoba-ku, Sendai, 980-8579, Japan
fatehi@abe.ecei.tohoku.ac.jp, abe@abe.ecei.tohoku.ac.jp

Keywords: Multiple modeling, Irregular self-organizing map, Adaptive identification, Neural network

Abstract

The identification method of *multiple modeling by the irregular self-organizing map* (MMISOM) neural network is presented, which improves the authors' previous method of MMSOM that uses the rectangular SOM. Inputs to the neural networks are parameters of the instantaneous model computed adaptively in each instant of time. The reference vectors of its output nodes are the parameters estimation of the multiple models. At each time, the model with nearest output to the plant output is chosen as the model of the plant. The irregular SOM used in MMISOM is a graph of all the nodes and some of the links that make a *minimum spanning tree* (MST) graph. It is possible to add new models if the number of models is initially less than the suitable one.

1 Introduction

The theory of *multiple models* (MM) started from the novel modeling method of a Gauss-Markov process by Magill [1] in 1965. In this type of modeling, which later was called *interactive multiple modeling* (IMM), there is a set of models for the process. The linear combination of their output gives the estimation of the original process. Middelton *et al*[2] revised Magill's method from interactive to *switching multiple modeling* (SMM) in which instead of the weighted combination of the models, only one of the models, which matches to the situation of the plant better than the others, is selected as the model of the plant.

Fatehi & Abe [3] introduced *multiple modeling by self-organizing map* (MMSOM) to compute and adapt the set of multiple models in SMM for a linear time-variant system. *Self-organizing map* (SOM) is a kind of unsupervised learning *neural network* (NN), which maps and classifies the characteristics of a collection of objects to a number of classes [4]. In MMSOM the parameters of a set of models for a plant are estimated using SOM and then the best one is selected for the present plant situation.

In this paper, *multiple modeling by the irregular self-organizing map* (MMISOM) is introduced to identify the multiple models for linear time-variant plants. The *irregular SOM* (ISOM) produces the local models. ISOM is a graph of all the nodes and some of the weighted links between them, such that they make a *minimum spanning tree* (MST) graph in which the integration of the weights is as small as possible. Using this type of network, it is possible to add new nodes. The algorithm does not initially know any thing about the suitable number of models for the plant and start with one or few models. It adds more models to accrue the desired accuracy. MMISOM shows more flexibility to cover the plant linear model space with fewer nodes than the previous MMSOM one.

2 ISOM neural network

2.1 Ordinary SOM neural network

SOM is an NN that learns its input and classifies them to some classes. The Euclidean distance between the present input vector value and the weights of the *reference vector* (RV) of each output nodes are computed. The network chooses an output node with the minimum distance as the winner node. The reference vectors of the winner node and its neighboring output node(s) are updated so that the RVs become closer to the present value of the input vector. This training procedure is applied for all of the input vectors.

The MMSOM algorithm [5] can make an MM for a linear time-variant plant, but there are some problems on using SOM for it. One of the problems is the existence of a lot of links between its nodes, which decreases its flexibility. To increase the flexibility, it needs to increase the number of nodes. Increasing the number of nodes means increasing the number of models and amount of computations. If an SOM with more flexible structure is used, it is possible to have accurate estimation without suffering of the number of the models.

The possibility of increasing or decreasing the number of nodes during the learning process is also important. If the number of nodes that was initially supposed for the SOM is not enough, we like to increase it without starting the algorithm from the initial step, something that is not possible in the rectangular structure SOM.

To solve the problem of flexibility, Kangas *et al* [6] introduced the ISOM neural network. Fortunately, it is possible to add or delete even one node to or from this network. ISOM has an MST graph. The MST graph and then the ISOM are explained in this section. Based on this network, the MMISOM algorithm is introduced in the next section and the method of increasing the number of models is presented.

2.2 ISOM network

Consider a network of n nodes. A weight L_{ij} is defined

for the link between every two nodes O_i and O_j. L_{ij} describes a distance measure between them. For this weighted network, the *MST graph* is defined as a connected graph of all n nodes and some of the links in the network, where the sum of the weights of its links is the minimum possible one. Replacing any of the links with another link(s), such that the graph keeps connected, does not decrease the sum of the weights of the links in the graph. The algorithm of finding an MST graph for a weighted network has thoroughly been explained on [7].

The *link weights* (LW) L_{jk} between the nodes of an ISOM is defined as the Euclidian difference between the RV of them. So, the link weight between the nodes O_j and O_k is

$$L_{jk} = \|w_j - w_k\|, \tag{1}$$

where

$$w_j = [w_{ij}]_i, \tag{2}$$

and w_{ij} is the reference vector weight between the input node I_i and the output node O_j. Using the above definition and the MST graph, the ISOM is defined as an SOM neural network with possible links between all nodes where the neighborhood of nodes are such that their links define an MST graph. If the RV of nodes change, LW between them also change. So, to keep the graph as an MST one, the neighborhoods of nodes may change as the networks trains.

3 MMISOM algorithm

3.1 An overview to the algorithm

Consider the dynamic plant $\mathcal{P}$ described by

$$y(t) = \varphi_{\mathcal{P}}{}^{T}(t)\theta_{\mathcal{P}}(t) + \omega(t), \tag{3}$$

where $\varphi_{\mathcal{P}} \in \mathcal{R}^{n_{\mathcal{P}}}$ is the regression vector of the plant, $\theta_{\mathcal{P}} \in D_{\mathcal{P}} \subset \mathcal{R}^{n_{\mathcal{P}}}$ denotes the time-variant parameters of the plant, ω is the white noise with mean zero and variance σ_ω^2, and y is the output. *The model parameter space* $D_{\mathcal{M}} \subset \mathcal{R}^{n_{\mathcal{M}}}$ is defined as the family of all the linear models $\theta_{\mathcal{M}}$ with the definition

$$y(t) = \varphi_{\mathcal{M}}^{T}\theta_{\mathcal{M}}, \tag{4}$$

for the plant (3) where $\theta_{\mathcal{M}} \in D_{\mathcal{M}} \subset \mathcal{R}^{n_{\mathcal{M}}}$ and $\varphi_{\mathcal{M}} \in \mathcal{R}^{n_{\mathcal{M}}}$ stand for the parameters and the regression vector of the model, respectively. The goal is to derive M linear local models

$$\Theta_{\mathcal{L}} = \{\theta^i \mid \theta^i \in \mathcal{R}^{n_{\mathcal{M}}}, i = 1,...M\}, \tag{5}$$

such that each θ^i is a good approximation for a space $d_{\mathcal{M}}^i \subset D_{\mathcal{M}}$, where $\bigcup_{i=1}^{M} d_{\mathcal{M}}^i = D_{\mathcal{M}}$. A model in $\Theta_{\mathcal{L}}$ is selected at each step of time as the model of the plant. This model is the best model for the plant using some defined criteria.

The irregular self-organizing map neural network is utilized in MMISOM to deal with the above multiple modeling problem. MMISOM algorithm is composed of three stages: 1) identifying the instantaneous models; 2) producing the bank of local models; and 3) selecting the best model, Fig. 2. At first stage, the input-output data of the plant is used to identify the instantaneous model of the plant in every instant of time. Any kind of known on-line identification method can be used to estimate the parameters of IM. This means sweeping $D_{\mathcal{M}}$ by producing $\theta_{\mathcal{M}}^o$ during the time, where $\theta_{\mathcal{M}}^o$ is the IM of structure $\mathcal{M}i$. The bank of *local models*, i.e. $\Theta_{\mathcal{L}}$, is produced by and embedded in an ISOM neural network. The number of inputs to the ISOM is equal to the number of estimation parameters in IM. It has M output nodes, where M is the desired number of the local models. IM is the input to the ISOM. The reference vector of each node of ISOM represents the local model, $\theta^i, i = 1,..., M$, for the plant.

The response of the plant to the control signal is compared with the responses of the local models to the same control signal. The model with the nearest response to the plant one is selected as the best model for the plant in that time. To achieve the advantageous of both past and present data, the following criteria is applied to measure the performance of the models:

$$J^i(t) = \beta|e_i(t)| + \gamma\left(\sum_{\tau} \alpha^{\tau} e_i^2(t-\tau)\right)^{1/2}, \tag{6}$$

where $J^i(t)$ is the performance measure for the model i at time t, $e_i(t)$ is the error between the plant output and the ith model at time t, α is the forgetting factor of the errors, and β and γ denotes the design parameters. J^i may define as the fitness of the model to the plant.

3.2 Increasing the models in MMISOM

Suppose a node with RV of w_* is going to be added to an ISOM with n node. At first, the difference between w_* and all other w_i, i=1,...,n are computed based on the Equation (1). The new neuron should connect to its nearest neuron in the present network, Fig. 1. So, the graph of the new ISOM is still MST. The new ISOM is the same as the previous one in old links and RVs. The only difference is in the new node and its link. Nevertheless, the links may change later during the training of the network.

A new model should be added to ISOM if *i*) none of the present models are suitable for the plant and *ii*) training the present network may defeat its ability to properly model the plant in some other learned situation. Therefore:

Condition 1: If the fitness of all of the models are more than the threshold a, a model should be added to ISOM.

The algorithm starts with few nodes and a node is added if it requires. Therefore the network is not over-noded. IM is the best choice for the initial value of the parameters and states of the new plant model. Training the ISOM

may stop for a while if the fitness of IM is more than the level c, to prevent training with wrong values. Training starts again when the fitness return to less than another level b. Therefore

Condition 2: The new model is added to the MMISOM models if the training is done.

If this condition is not satisfied, adding the new model is postponed until it does.

3.3 Identification example

The MMISOM algorithm is used to model a linear time-variant plant

$$y(t) = -a_1 y(t-1) - a_2 y(t-2) + b_0 u(t) + b_1 u(t-1) + \omega(t), \quad (7)$$

where ω is a white noise with mean zero and variance 0.2. u is supposed to be a random signal with uniform distribution $\mathcal{U}(-1,1)$.The four parameters of the plant (7) are supposed to get each of the 5 following sets

$$\begin{aligned}(a_1,a_2,b_1,b_0) \in prmtrset = \{&(-1.1\ ,0.24,0.05,0.005)\\ &(-0.9\ ,0.14,0.65,0.455),(-0.7\ ,0.06,0.45,0.225),\\ &(-1.3\ ,0.36,0.25,0.075),(-1.48,0.49,0.85,0.765)\}.\end{aligned} \quad (8)$$

When one of the above sets is assigned for the plant parameter, it remains for 1000 steps. After that one of the other sets of the *prmtrset* is randomly chosen.

IM is computed by the RLSE identification [8] with forgetting factor rate 0.98. The algorithm is simulated for 60,000 steps. The changes in plant parameters are shown in Fig. 4 together with obtained IM one. The speed of convergence of IM parameters depends on λ and the natural frequencies of the plant.

To apply the MMISOM algorithm, the initial network is supposed to have just one node. The learning factor of the network is set to 0.01 and the neighborhood is one for whole of the algorithm. The neighborhood is checked every 30 steps to keep it as an MST graph. The criterion of Equation (6) is used with $(\alpha,\beta,\gamma) = (0.95,1,0.2)$ and for the Condition 1 and 2 $(a,b,c) = (1,0.2,0.2)$. The order of selection of the plant parameters from the *prmtrset* repeats after 20,000 steps to illustrate any improvement in the identification of the plant.

Fig. 4 illustrates the parameter of the BM in addition to the plant and IM ones. In early stages of modeling, BM can get the plant by addition of models after IM has already reached it. But gradually, BM approaches to the plant faster than IM, as is found out by comparing the results of Fig. 4.

The fitness of IM and BM are also illustrated in Fig. 4. The fitness of BM is higher than IM one in early steps but it is lower during the transient periods after all of the changes from step 15000, except of the change in the step 26001. A node is added after the change in the plant parameter if the BM fitness is more than IM one. Fig. 3 illustrates the structure of the ISOM at the end of simulation, which has 12 nodes.

4 Conclusion

The MMISOM has been presented, which is a method to obtain a set of models for different situation of a plant using the irregular self-organizing map neural network. The suitable number of nodes in ISOM needs not to be known at the beginning of the algorithm when MMISOM is used. MMISOM shows to have more flexibility than the previous method MMSOM also presented by the authors.

Simulated results shows that it can introduce the model of the plant much faster than the adaptive IM one after the ISOM has been sufficiently trained. Training of ISOM is on-line. In fact MMISOM can introduce a model of the plant in a situation after it has been experienced for a few times. Therefore, none of the models are suitable at the early steps and before enough training. Fortunately, The algorithm gains an adaptive identifier, which can act as the best model if it is better than all of RVs.

MMISOM is specifically suitable for the plants that its parameter changed abruptly. As an example a robot arm that should handle different loads one after another.

References

[1] Magill DT (1965), Optimal adaptive estimation of sampled stochastic process. IEEE Transaction on Automatic Control AC-10(4):434-439.

[2] Middleton RH et al (1988), Design issue in adaptive control. IEEE Transaction on Automatic Control 33(1):50-58.

[3] Fatehi A, Abe K (1999), Plant identification by SOM neural networks. European Control Conference, Karlsruhe Germany, Aug 31-Sept 3, 1999, F190.

[4] Kohonen T (1997), Self-Organizing Maps. Springer-Verlag, London England.

[5] Fatehi A, Abe K, Multiple modeling of time-varying systems by self-organizing map neural network. Submitted to the International Journal of Adaptive Control and Signal Processing.

[6] Kangas J, Kohonen T, Laaksonen J (1990), Variants of self-organizing maps. IEEE Transactions on Neural Networks 1(1):93-99.

[7] Sedgewich R (1988), Algorithms, 2nd ed. Addison-Wesley, Reading Massachusetts.

[8] Ljung L (1983), Theory and practice of recursive identification. The MIT Press, Cambridge Massachusetts.

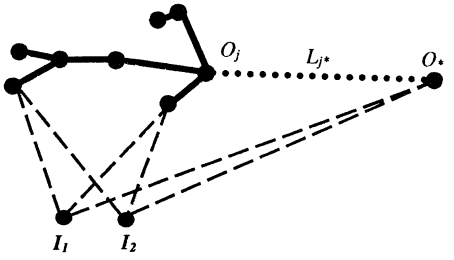


Fig. 1) O_* is added to the ISOM.

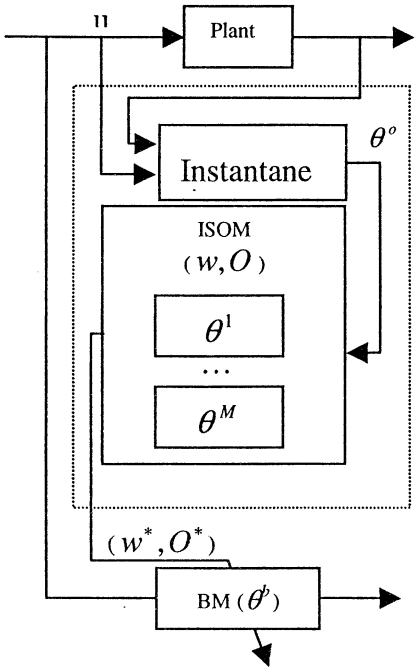


Fig. 2) Structure of the MMISOM

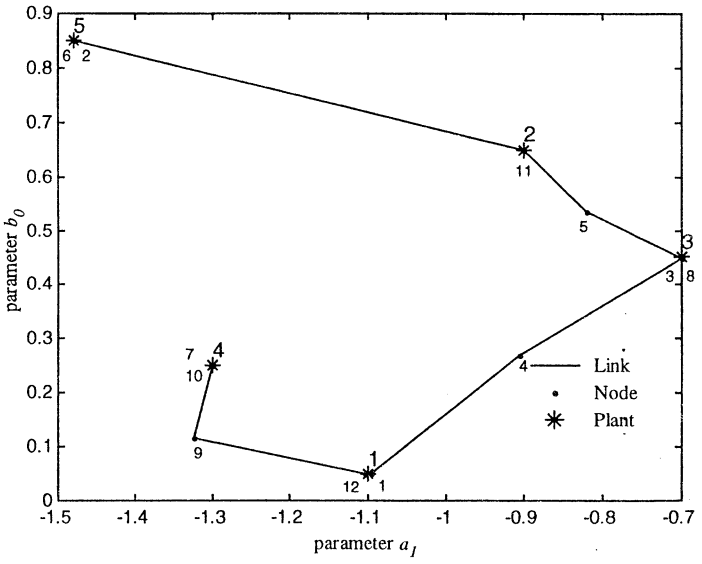


Fig. 3) The structure of SOM in example of Section 2.5 in the step 60,000. Only two parameters are shown.

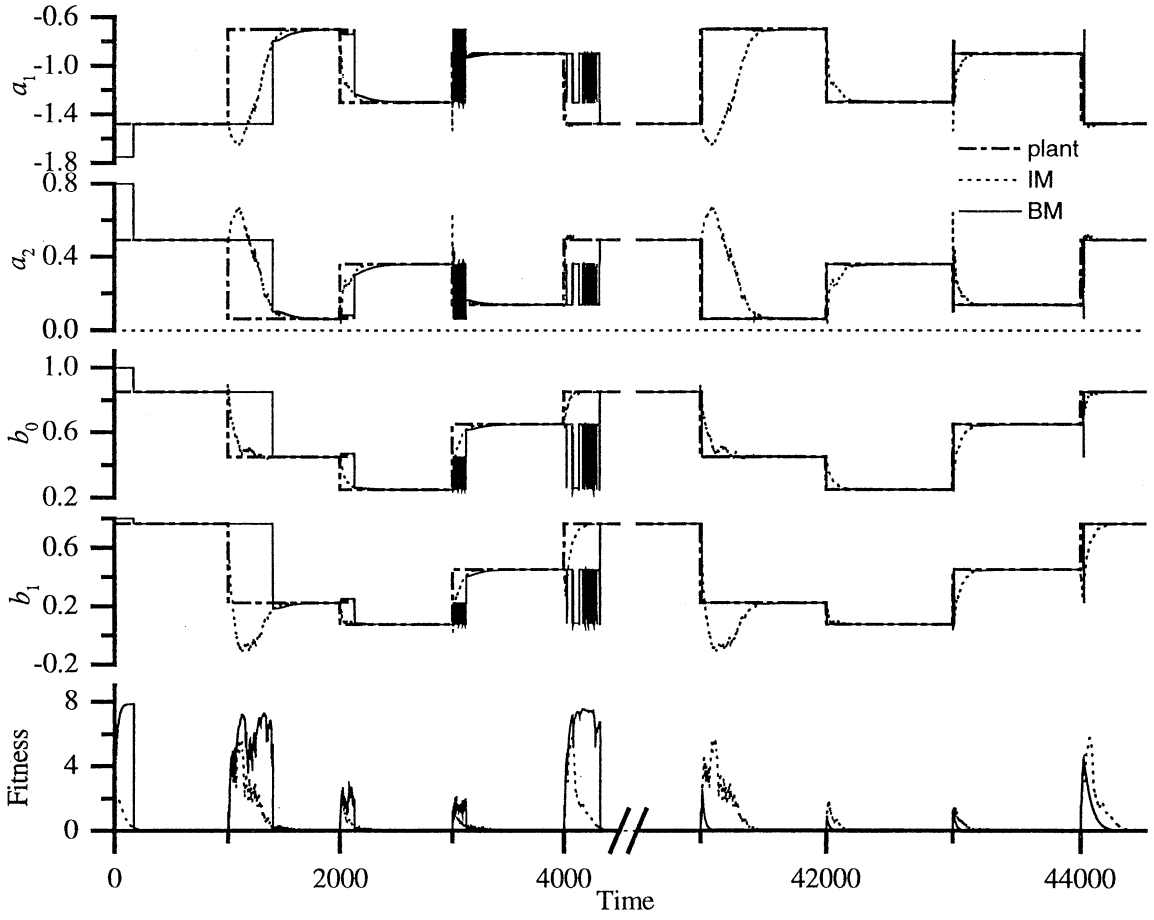


Fig. 4) The parameters of the plant, IM and BM.
The plant parameters are the same in both portions.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

An *in vivo* approach to molecular computation

○ Kentaro Matsuno, Masahito Yamamoto, Toshikazu Shiba and Azuma Ohuchi

Graduate School of Engineering
Hokkaido University
Sapporo 060-8628, Japan

E-mail address : {kentaro, masahito, shiba, ohuchi}@dna-comp.org

Abstract

We have been trying to design an *in vivo* computing approach and achieve it based on biological experiments. It aims to make good use of gene expression and protein synthesis in our approach. Concretely, we produced cells that correspond to the Boolean function; that is, x returns the output of 1, 0, and $\bar{x}$ returns the output of 0, 1, to the input of 1, 0, respectively. In biological implementation, we chose *Escherichia coli* as a cell and carried out a recombinant DNA experiment to implement the function. An inducing substance in the environment of the cell was used as the input, and the detective protein synthesized in the cell was used as the output.

Keywords: Molecular computation; Biological computation; In vivo approach

1 Introduction

Molecular computation and DNA computation have been proposed as ways of solving certain types of computational problem. Most researches have suggested that problems can be encoded into DNA and that computational operations can be performed *in vitro* (such as in a test tube, Adleman [1]). Many papers have recently pointed out the possibility of *in vivo* (in living cells) computation (Smith [7], Ji [4] and Weiss *et al.* [8], [9].) Particularly notable is the proposal by Eng [2] of an algorithm for solving a 3-conjunctive-normal-form Satisfiability problem *in vivo*. Gardner et al. [3] constructed a genetic toggle switch that forms a cellular memory unit in *E. coli*.

DNA computation is the most general field of molecular computation; however, much interest has been shown in extracting computational power of coupled chemical reactions in living cells. It is expected that problems such as scalability and the high cost of *in vitro* approaches can be overcome by using *in vivo* approaches, and the technology has implications for drug and biomaterial manufacturing. However, there have been few theoretical and experimental studies on *in vivo* approaches. We have therefore tried to design an *in vivo* computing approach and realize it based on biological experiments. Then we aim to make good use of gene expression and protein synthesis in a cell.

2 *In vivo* approach

First, the gene expression mechanism used in this paper is described briefly. A structural gene that encodes a protein is restricted by binding of a particular repressing protein to the upstream regulatory region. However, if a particular inducer is present, the structural gene is expressed in a cell because the repressor is inactivated by the inducer.

Our *in vivo* approach to computation is to produce cells that correspond to the Boolean function; that is, x returns the output of 1, 0, and $\bar{x}$ returns the output of 0, 1 to the input of 1, 0 respectively (See Table 1). We adopt the input of 1 or 0 depending on whether an inducing substance is present or not in the environment of the cell (input substance + or -). The output of 1 or 0 corresponds to synthesis or non-synthesis of the protein affected by the inducer (output protein + or -). In this paper, we call the cells that correspond to x and $\bar{x}$ cells "T" and "F", respectively (See Table 2).

	Input	Output
x	1	1
	0	0
$\bar{x}$	1	0
	0	1

Table 1: Input and output of x and $\bar{x}$.

Cell	Input substance	Output protein
T	+	+
	-	-
F	+	-
	-	+

Table 2: Cells "T" and "F".

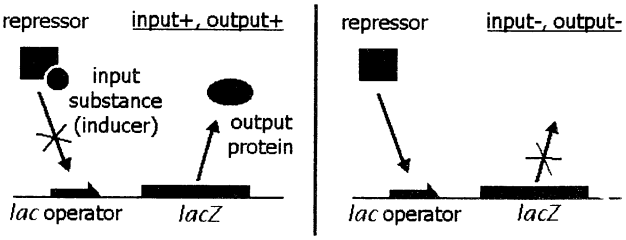


Figure 1: The two ideal cases for cell "T", in the presence or absence of an input substance (inducer). Case 1: input +, output +.; Case 2: input -, output -.

3 Biological implementation

In *biological implementation*, we chose *E. coli* as a cell and carried out a recombinant DNA experiment to implement the functions. Then, they were implemented on *E. coli* plasmids, the biological detail is as follows. The synthetic inducer, isopropyl-β-D-thiogalactopyranoside (IPTG), was added to the environment of the cell as an input substance. The inducer IPTG inactivates the repressor that binds to the textitlac operator region. β-Galactosidase synthesized from *lacZ* structural gene was chosen as the output protein to distinguish the expression of the gene. The output protein is chosen so that it results in a detectable phenomenon. In this respect, β-galactosidase is easy to assay. In short, "input substance +" in table 2 means that IPTG is added to the environment of the cell, and "output protein +" means that β-galactosidase is synthesized from a reporter gene.

3.1 Cell "T"

Cell "T" is expected to act as follows. If IPTG is present, β-galactosidase is expressed in *E. coli*. In contrast, β-galactosidase will not be expressed in the absence of IPTG. Since *E.coli* has this property by nature, we chose *E.coli* CF1648 as cell "T". This strain of *E.coli* has wild-type *lac* operator region and *lac* structural genes. Figure 1 shows the two ideal cases for cell "T". A *lac* genes are normally repressed by the *lac* repressor, a product of the *lacI* gene. It should be possible to detect β-galactosidase only when IPTG is present, since IPTG inactivates the *lac* repressor. In the absence of IPTG, β-galactosidase is not synthesized in CF1648.

3.2 Cell "F"

Cell "F" is expected to act as follows. If IPTG is present, β-galactosidase will not be expressed in *E. coli*. In contrast, β-galactosidase is expressed in the absence of IPTG. This is not a natural property of *E.coli* and can only be implemented with genetic recombination. We chose *E.coli* JM109 as cell "F". We transformed the property of *E.coli* JM109 by introduction two plasmids, pNF20 and pNF265, were introduced in *E.coli* JM109. These plasmids contain the promoter region and coding region of *nfxB*, respectively (See Figure 2). NfxB protein binds to the *nfxB* operator and behaves as a negative autoregulator

Figure 2 shows the two ideal cases for cell "F". The following is the function of the plasmids pNF20 and pNF265 in *E. coli*. In the presence of IPTG, the *lac* repressor is inactivated, and NfxB protein is expressed under the *lac* promoter in pNF20. Then the NfxB protein binds to the *nfxB* promoter region of plasmid pNF265 and regulates the expression of the *lac* gene under the *nfxB* promoter. If NfxB protein is not synthesized in pNF20 in the absence of IPTG, β-galactosidase is produced from the *lacZ* gene in pNF265.

4 Readout of output protein

After producing cells "T" and "F", we carried out a readout experiment on these cells. Since the output of 1 or 0 corresponds to synthesis or non-synthesis of the protein affected by the input substance is synthesized in *E. coli*, the output protein β-galactosidase in the presence or absence of the input substance must be

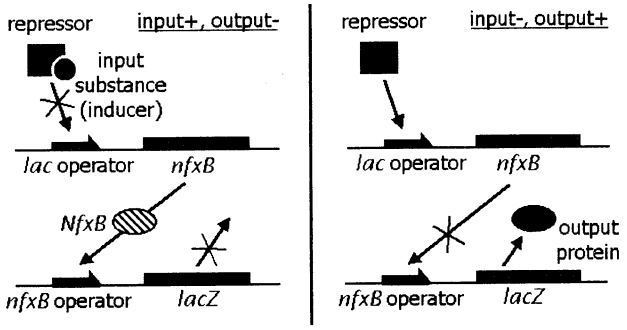


Figure 2: The two ideal cases for cell "F", in the presence or absence of an input substance (inducer). Case 1: input +, output -.; Case 2: input -, output +. The plasmid pNF20 contains the upper fragment, and the plasmid pNF265 contains the lower fragment.

detected. Expression of *lac* was measured in terms of β-galactosidase activity.

Cell "T" (CF1648) was grown in 5 ml of LB medium to optimal density at 600 nm of 0.2. containing 250 μg of 40 % glucose. Cell "F" (JM109) was also grown in 5 ml of LB medium containing 25 μg of ampicillin per ml, 5 μg of chloramphenicol per ml, and 250 μg of 40 % glucose. Both of these strains were induced with a final concentration of 0.25 mM IPTG or not. After incubation for a further 2 h, 10 μl of toluene was added to 900 ml of buffer Z (40 mM NaH_2PO_4, 60 mM Na_2HPO_4, 10 mM KCl, 1 mM $MgSO_4$, 50 mM β-mercaptoethanol) and 100 ml of each sample. Toluene was evaporated through incubation for 1 h. After preincubation in a water bath at 28 ℃, 200 μl of o-nitrophenyl-β-D-galactopyranoside was added to each sample (*start*). Then 500 μl of Na_2CO_3 was added at the optimal density at 420 nm of 0.2 - 0.8 (*stop*). The reaction time (from *start* to *stop*) was recorded. Finally, measured the optimal densities at 420 nm and 550 nm were measured using a spectrophotometer. β-Galactosidase activity was analyzed quantitatively as follows:

$$\beta - galactosidase\ activity = \frac{A_{420} - (1.75 \times A_{550})}{T \times V \times A_{600}} \times 100.$$

A_{420}, A_{550} and A_{600} are optimal densities at 420 nm, 550nm and 600nm, respectively, and T and V are time (min) and sample volume (ml), respectively.

Figure 3 shows β-galactosidase activity of each strain as indicated, i.e., CF1648 as "T" with IPTG (hatched bar) and without IPTG (solid bar); JM109 (containing pNF20 and pNF265) as "F" with IPTG (open bar) and without IPTG (dotted bar). The figure shows the following four cases. In cell "T", β-galactosidase was synthesized in the presence of IPTG and it was not synthesized in the absence of IPTG. In cell "F", β-galactosidase was not synthesized in the presence of IPTG and it was synthesized in the absence of IPTG. In order to distinguish between the output of + and -, a β-galactosidase assay was carried out.

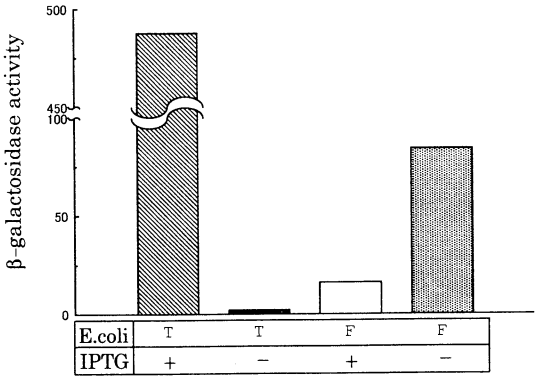


Figure 3: Effect of β-galactosidase assay.

5 Summary

In this paper, we described an *in vivo* approach to molecular computation based on gene expression and protein synthesis. Cells that corresponded to the Boolean function were produced by genetic recombination. It was able to be confirmed whether the output protein affected by the input substance was synthesized (i.e., whether the output was + or -) by means of a β-galactosidase assay.

From the point of view of computation, the present *in vivo* computation is not beyond *in vitro* DNA computation, much less an electronic computer. However taking into consideration the relationship to the field of biochemical and drug manufacturing, *in vivo* computation seems to be a promising approach.

Methods and materials

Bacterial strains and plasmids

E.coli JM109 was used as the host strain for the plasmids pNF20 and pNF265. *E.coli* CF1648 is a wild-type strain.

pNF20 was provided by Shiba [6], and which carries the entire region on the *nfxB* gene on pSU2719, a derivative of pACYC184, was constructed as follows. A *Bam*H I fragment of pNF3B was ligated into the *Bam*HI site of pSU2719. In the resulting plasmid, pNF20, the reading frame of the *nfxB* gene was placed in the same orientation as that of the *lac* gene.

pNF265, was provided by Shiba [6], harbored the 272-bp fragment in frame. The DNA fragment (272 bp) containing the promoter region and part of the coding region on *nfxB* was isolated from pNF3E, and the isolated by *Eco*R I digestion followed by *Ava* II digestion and polymerization with T4 DNA polymerase. The fragment was ligated with pMC1403, which had been digested with *Eco*R I and *Sma* I .

β-galactosidase assay

E. coli CF1648 and JM109 carrying appropriate plasmids were cultured to an optical density at 600 nm of 0.2, and β-galactosidase activity in the cells was measured as described by Miller [5].

References

[1] Adleman, L., "Molecular computation of solutions to combinatorial problems" *Science* 266, 1021-1024, 1994.

[2] Eng, T. L., "On solving 3CNF-satisfiability with an in vivo algorithm" BioSystems 52, p. 135-141, 1999.

[3] Gardner, Timothy S., Charles R. Cantor, James J. Collins, "Construction of a genetic toggle switch in escherichia coli" *Nature* 403, p. 339-342, 2000.

[4] Ji, Sungchul, "The cell as the smallest DNA-based molecular computer" *BioSystems* 52, p.123-133, 1999.

[5] Miller, J. H., "Experiments in molecular genetics" Cold Spring Harbor Laboratory, Press, Plainview, N.Y., 1972.

[6] Shiba, T.,K. Ishiguro, N, Takemoto, H, Koibuchi, and K.Sugimoto, "Purification and Characterization of the *Pseudomonasaeruginosa* NfxB Protein, the Negative Regulator of the *nfxB* Gene" Journal of Bacteriology, p. 5872-5877, Oct. 1995.

[7] Smith, Warren D., "DNA computers in vitro and vivo" DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol. 27, 1996.

[8] Weiss, Ron, George Homsy, "Toward *in vivo* Digital Circuit" Dimacs Workshop on Evolution as Computation (Princeton, NJ), January 1999.

[9] Weiss, Ron, Thomas F. Knight, Jr., "Engineered Communication for Microbial Robotics" In Proceedings of the 6^{th} International Meeting on DNA Based Computers, held at Leiden University, June 13-17, 2000.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Music and Meta-nature

A short survey of alife music

Rodney Berry
ATR Media Integration and Communications Laboratories
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, Japan.
rodney@mic.atr.co.jp
www.mic.atr.co.jp/~rodney

Keywords: artificial life, music, art, virtual worlds

Abstract: Throughout history, musicians have tried to harness the dynamic properties of natural processes to create music. Wind-chimes and Aeolian harps play music from the unpredictable wind. 20th century artists and composers have entered into a collaborative dialogue with nature on many levels to make music. Our response to music comes from our evolved response to nature but now informs our response to nature as a generator of musical meaning. Musicians and composers, inspired by the creative potential of Artificial Life and related fields of study, are also working to translate the fruits of science into musical compositions and creative systems. I look briefly at the works of several composers in the field of alife music and invite the reader to explore examples to be found on the web.

The natural world provides inspiration for composers and musicians. Usually the sounds heard in the environment are abstracted in some way by the musician and re-fashioned into musical phrases made with the voice or some kind of instrument. There is another musical tradition however in which the hand of the musician is replaced by some natural process or force. The most common example of this is the wind-chime, an instrument that plays a usually limited and undetermined series of notes depending on the air turbulence around the instrument. Another such instrument is the wind harp or *aeolian harp*, an instrument consisting of several wires loosely stretched across some kind of sounding box or resonator. When the wind blows across the wires, some of the middle to upper harmonics of the vibrating body can be heard. In both these examples, there is no human intent involved in the actual music production. The maker of the instrument has essentially fashioned a system, a set of interacting elements by which music is produced. After choosing the organization of these elements, the maker has no further control over the sound produced. The rhythms, dynamics and melody are determined by the patterns of turbulence of the wind from moment to moment. Still, a listener can perceive this as music. Something about the way we hear sound and respond to it leads us into some form of abstraction about it. Music is often described as *organized sound*. This organization takes place both when it is produced and when it is heard. The giver and the receiver of the musical message each take part in this organization process.

We respond emotionally to music because our sense of wellbeing is connected to our survival and our perceived relationship to our immediate environment. It appears that our brain has evolved to look for qualitative changes over time and to recognise quite complex patterns in these changes. We use these patterns as a way to try to predict what our environment will do from one moment to the next. When we find a pattern in the stimuli presented to us, and the pattern is consistent with other stimuli and with what we already know about the world, we feel relaxed. If there is uncertainty or conflict, we experience unease and discord. My guess is that the abstracted sound patterns found in music connect with our instinctive desire to order and model our world. Composers exploit this by setting up expectations in the listener, then, either fulfilling or frustrating those expectations over a period of time. If abstract sonic patterns activate our evolved mechanisms for perception and cognition of our world, then we can learn to attach significance to this as a form of communication between humans. Once we do this, we begin to listen to music for its own sake as a part of our symbolic environment as social communicating beings. This is separate from our experience of sound as an environmental information source. We respond to musical sound as if it comes from the real world.

When we go into a natural environment, such as for a walk in the forest, we carry with us all the learned responses from our musical culture. We then find ourselves responding to real sounds as if they were abstract musical sounds. When we hear the sound of the wind in the trees, the sounds of birds and insects, the sound of running water, it is not hard to believe we are listening to a very complex piece of music with its own formal structures waiting to be decoded. Our perception of some kind of underlying order produces an aesthetic response to that order or pattern, not just with sound but with visual stimuli as well.

Artificial life gives humans a chance to experience another kind of nature capable of enriching our hungry sensorium. Our faculty for internal organization and aesthetic appreciation of incoming stimuli can be applied to artefacts of computer driven

processes too. Artists such as Karl Simms[1] have successfully applied the pattern making properties of artificial life software to visual art. Simms has used various forms of evolution by selection to produce pleasing visual images. Others have quickly recognised that similar things can be done with music. In this paper, I will look at some work by a small number of composers who make music using artificial life. I also provide some web links to sites, some containing software and musical examples that readers can try for themselves. It is my intention to maintain a set of web links accessible from my home page that will expand as this area of musical experimentation expands.

Firstly, depending on one's own definition of artificial life, much of the field of algorithmic composition by computer could be dealt with here. For the sake of brevity, I would like to concentrate on work produced using algorithms that seek to emulate systems and processes from nature.

Cellular automata are a branch of alife that produces rich patterns over time and have been explored by a number of composers. A pleasing pattern on a computer screen does not guarantee a pleasing pattern to the ear. The point where the programmer/mathematician becomes an artist is the point where one determines the mapping from one parameter to another to produce a musically interesting result. For example, supposing we make an algorithm that produces a pretty pattern, we decide that the vertical axis will represent pitch and the horizontal axis will represent time. Then the pattern is read from left to right and musical events are scheduled according to their placement on the screen. In this way, there might be some kind of intuitive link between what we see on the screen and what notes are played by whatever device finally makes the sound. The sound produced by such a system can sometimes be interesting but somehow not usually engaging. Eduardo Reck Miranda has produced many musical pieces using cellular automata. His software system CAMUS[2], (Cellular Automata MUSic) goes well beyond this simple kind of mapping to produce a rich result. CAMUS uses two cellular automata to generate patterns for output as MIDI. Firstly, Conway's LIFE automaton is used to determine the pitch material. Instead of the simple mapping I suggested earlier, Miranda lets the horizontal position of a cell determine the lower interval of a musical *triple* or *triad* (a vertical grouping of three pitches), then the vertical position determines the upper interval. Then the states of the neighbouring cells are used to determine the duration and articulation of the notes referred to by the cell. To determine what instrument plays each note, a version of David Griffeath's Demon Cyclic Space is used. Each state of the demon automaton selects between 4 instrument sounds. These two layers run concurrently to generate music.. I believe that Miranda's choice of mapping works well because it produces a qualitative change based on the groupings and movements of these groupings over time rather than just a simple linear change in pitch. Miranda has made a 3D version of this system and a cellular automata powered granular synthesiser called Chaosynth[3]

Artificial evolution is another aspect of artificial life being explored by musicians. Mostly inspired in one way or another by the work of Karl Simms, a number of composers have developed systems for evolving musical patterns through a process of selection. GenBebop[4, 5]is a project directed by Lee Spector using genetic programming to produce programs for interactive jazz improvisation. His software analyses midi data provided by a human musician and plays back answering phrases on a synthesiser. An automated *critic* function attempts to improve the responses over time. The challenge in this case is to find a way of improving the critics (who will watch the watcher?). A Program called GenJam[6] by David Biles explores a similar *call and response* method of music making.

Placing a human in the loop at each iteration of an evolutionary process becomes tedious to say the least. Some have tried getting a number of people to vote and thus share the burden7. To avoid this dehumanising tedium, ways are being searched out to automate the evaluation process. Jonatas Manzolli [8] has put a lot of energy into the development of musical fitness functions to remove the bottlekneck posed by excessive human intervention. In Vox Populi, a program for the Windows platform, he divides musical fitness into three sub-functions, Melodic, Harmonic and Voice Range fitness. These, combined with a function that can identify a group of notes' relationship to a tonal center, form a buffer between the user and the countless iterations of evaluations normally required in conventional human-centered *aesthetic selection*. Manzolli's software also gives the user a visual representation of aspects of the music to provide another mental model of what is going on in the music.

Another approach to evolving musical structures is to create interactive virtual environments where artificial musical creatures can interact with each other in real time and evolve potentially free of the need for user intervention. Composer Palle Dahlstedt and complex systems researcher Matts Nordahl made such a world called *Living Melodies*[9, 10] The inhabitants of this world inherit both musical and physical behaviours from their parents. All their behaviours both musical and 'physical' are described in a genome resembling a computer program. Creatures play notes to attract a mate (just like real musicians!) They only make sounds when they hear sounds that they like. A variety of rules cause a variety of interesting behaviours to evolve and many parameters can be set by the user to produce radical changes in the sound that comes out. Sometimes populations will swell to plague proportions filling the world then some will die out and some other mutant strain will take over. The interesting thing about listening to and interacting with Living Melodies is that the music becomes a history of a population's

growth and demography. We are accustomed to music following a narrative curve of some kind so it is easy for an audience to accept this kind of work as music.

My own work in this area also involves the making of virtual worlds populated by alife creatures. My older work, Feeping Creatures[11] is an artificial environment that is navigated as a three dimensional world from a first person perspective. The user hears sound all around as they move around the landscape. Creatures mate according to levels of consonance or dissonance between their musical chromosomes. The more recent Gakki-mon Planet[12] uses a more complex chromosome to make elaborate behaviours and music of varying timbre, pitch and rhythm. The user can use a midi keyboard to influence the creatures while exploring their environment.

In conclusion, I believe that alife can enrich a composer's repertoire of musical tools by providing patterns of musical information of a variety not easily made with mere random or chaotic processes. Composers like the ones briefly described are charting a new creative territory where authorship gives way to a form of co-operation or husbandry between a human and a computer. I urge the reader to visit my web page where I maintain a list of links to the music of artificial life and related areas of artistic interest. This page can be found from my homepage:

http://www.mic.atr.co.jp/~rodney

---

[1] www.genarts.com/karl/

[2] *http://website.lineone.net/~edandalex/camus.htm*

[3] *http://website.lineone.net/~edandalex/chaosynt.htm*

[4] *http://hampshire.edu/~lasCCS/genbebop.html*

[5] *see also: Spector, L., and A. Alpern. 1994. Criticism, Culture, and the Automatic Generation of Artworks. In Proceedings of the Twelfth National Conference on Artificial Intelligence, AAAI-94, pp. 3-8. Menlo Park, CA and Cambridge, MA: AAAI Press/The MIT Press.*

[6] *http://www.it.rit.edu/~jab/GenJam.html*

[7] *There once was web project took votes from all who clicked on the site to evolve the music. It has unfortunately disappeared but I hope to locate and mirror it so watch my links page...*
see also: Elenora Bilotta, Pietro Pantano and Valerio Talarico Synthetic Harmonies: an approach to musical semiosis by means of cellular automata . 7th Conference on Artificial Life, p537-546

[8] *http://www.itaucultural.org.br/invencao/papers/Manzolli.htm*

[9] *http://www.design.chalmers.se/palle/living-melodies/*

There is a Windows version you can download and use.

[10] *Palle Dahlstedt and Matts Nordahl. Living Melodies: Coevolution of Sonic Communication - Proceedings of the First Iteration Conference on Generative Processes in the Electronic Arts, Melbourne, Australia, December 1-3 1999*

[11] *http://www.cofa.unsw.edu.au/research/rodney/FC_table.html*

[12] *Described in more detail in Dr Ryohei Nakatsu's session on human communication with alife and robots at this conference.*

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Learning Motion Coordination Algorithm for Distributed Agents in Cellular Warehouse Problem

○Katsumi Hama*, Sadayoshi Mikami**, Keiji Suzuki** and Yukinori Kakazu***

* Hakodate College of Technology
14-1 Tokura-cho, Hakodate,
Hokkaido 042-8501, Japan
E-mail:hama@hakodate-ct.ac.jp

**FutureUniversity-Hakodate
116-2 Kamedanakano-cho,
Hakodate, Hokkaido 041-8655,
Japan

***Hokkaido University
Kita-13, Nishi-8, Kita-ku, Sapporo,
Hokkaido 060-8628, Japan

Abstract

An approach to coordinate motions of transport tables for cellular warehouse problem is shown. In the proposed approach, the tables are considered to be autonomous agents, and a built-in behavior function given by ANNs and the evolved problem-oriented connection weights navigate the agents to their specified goals. To determine the agent to be moved, a measure of the priority to move is introduced. We show that a distributed agent with the learned behavior function and the negotiation value performs a similar strategy to a "serializable" solution in N-puzzle problems.

Keywords: Conflict Resolution, Action Selection, Artificial Neural Networks, Evolutionary Programming

1 Introduction

This paper describes an approach to coordinate motions of distributed learning agents for cellular warehouse problem. The warehouse is a collection of two dimensional cells. Each table is considered as an autonomous agent and can move in orthogonal four directions. According to the requests for loading and unloading, an agent must either move to its goal or to a free space to spare a room for other agents.

The solution of the problem is a motion sequence of agents. To attain system's robustness, we assume a N-puzzle-like control strategy with autonomous sensor-based behavior and minimum communication. Behavior function of each agent is given as ANNs (artificial neural networks) and the problem-oriented connection weights are evolved by Evolutionary Programming [1].

The activation/inactivation is controlled by a negotiation protocol. The protocol solves deadlock and reactivation problems that arise during local interactions among neighboring agents. Since the final goal is attained through each agent's goal-directed actions and interactions for resolving these motion conflicts, we need to give a measure of "mobility" to each agent with respect to the given problem. The measure introduced here is used to prioritize agents' motions and we call it a "motivation value" [2]. This ensures that the agent with bigger motivation value has a chance of taking its motion determined by its behavior function.

For the reactivation problem, we introduce a simple transfer mechanism of motivation value [3]. Initially given motivation value is reduced according to a number of unit motions made by an agent. If a motion is blocked by another agent, the request value to move with a "request-to-move" signal is transmitted to the agent. When an agent has attained its own goal, remaining motivation value is spread and shared among the rest of agents. Usually the bigger the request value to move becomes, the more easily reactivation behavior will occur. However, there is a problem-specific tradeoff between the request to move for an agent and the activation of its own motion, which is the target of the proposing approach.

In N-puzzle problems, it is known that a series of subgoals that forms a wall-like structure gives an efficient and unique solution to the problem. This is called a "serializable" solution [4]. Through simulations, we examine a motion sequence like serializable solution.

2 Serializable solution and problem solving

Each agent in cellular warehouse problem becomes one of three states according to its situation: i.e. active, inactive (stable) and free. Given an initial (a current) state and a goal state, the solution of this problem is to obtain a motion sequence to goal attainment.

If we consider the goal of each agent in warehouse problem as a subgoal, serializable subgoals will be an effective solution like in N-puzzle problem. Serializable subgoals mean there exists an ordering among the subgoals such that they can always be solved sequentially without ever violating a previously solved subgoal in the order. Therefore, for example, 8-puzzule subgoals of positioning the tiles one at a time are not serializable since once the first two tiles in a row have been placed, in general they must be moved in order to place the third. It has been pointed out, however, once the entire top row is correctly positioned, the remainder of the puzzle can be solved without distributing the top row. Thus, even though the individual subgoals are not serializable, they can be grouped together into blocks which are serializable.

Fig.1 shows the outline for problem solving. The

agent to be moved is determined according to the result of local interaction, while a motion of the corresponding agent is decided by the behavior function. All agents have the same behavior function that is a single network in the network population. Since, with no prior knowledge, it is difficult for agents to execute various kinds of task, we first make them learn fundamental motions (e.g. straight, rotate right, left etc.) using the training set of small size problems. By repeating this training starting from the simplest problem, problem-oriented networks are constructed incrementally in the network population. After training, the population is adopted as that of the initial state to solve a given task. If the population cannot obtain the solution of the task, we go on with more supplementary training for it.

3 The Approach Method

3.1 The Behavior Function

Behavior of agents is given by ANN. Since a time-dependent process is necessary for acquiring motion sequences of agents, we use the recurrent neural network which has the ability to memorize previous states. To find problem-oriented neural networks, we adopt the fixed structure including recurrent loop connection (no with inter-unit connection) within hidden layer and its connection weights are evolved through an evolutionary learning. The delayed signal will be utilized as memory of hidden layer unit output.

To receive inputs from the environment, the agents have binary (on/off) sensors for static obstacle (includes wall), mobile obstacle (includes agents), vacant cell, goal direction and motion request. Each sensor acquires states in surrounding orthogonal four directions. The goal direction can have multiple inputs simultaneously. Motor output is move to orthogonal four directions.

3.2 Evaluation Criteria and Evolutionary Operators

Evaluation criterion of the network performance for the obtained motion sequence is given as follows:

$$E = c_1 \cdot DIST + c_2 \cdot \left(NC_{wall} + NC_{object} + NG_{motion}\right) + c_3 \cdot DIF + c_4 \cdot N_{nmove} \quad (1)$$

$$DIST = \sum_{i=1}^{NA} \left(\left| XN_g^i - XN_c^i \right| + \left| YN_g^i - YN_c^i \right| \right)$$

$$DIF = \sum_{j=1}^{NO} \left(OUT_{ave} - OUT_j \right)^2$$

where

NA : the number of agents

XN, YN: x and y coordinates in two dimensional cellular space (suffix g and c indicate goal state and current state and i is the agent number)

NO : the number of motion directions

OUT_j : the number of output from j-th motor

NC_{wall} : the number of collision against wall

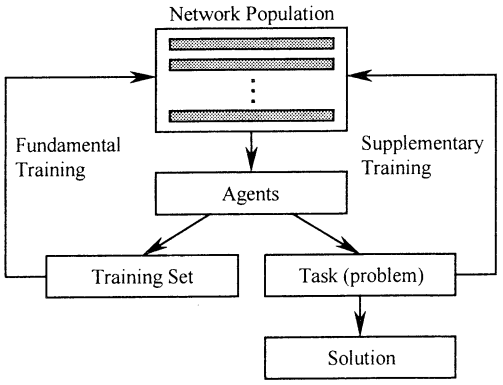


Fig.1 The outline for problem solving

NC_{object} : the number of collision against other agents

NG_{motion} : the number of unit motions made until the goal state is attained

N_{nmove} : the number of unit motions with no move

c_1, c_2, c_3, c_4 : constants

The first term of E evaluates 'distance' to given goal state in the final state, the second term penalizes ineffective motions, the third term penalizes bias toward certain motor output and the fourth term penalizes motions with no move.

Using the evaluation criteria, we calculate fitness function of the i-th network in the network population;

$$F_i = \frac{E_{\max} - E_i}{E_{\max} - E_{\min}} \quad (2)$$

and, according to this, the networks above average performance are inserted to the population after the removal of low performance networks from it.

The mutation operation (changes in connection weight) will be given according to

$$w = w + c \cdot N\left(\mu, \sigma^2\right) \quad (3)$$

$$c = \begin{cases} \dfrac{F_{\max} - F_i}{F_{\max}} & \left(n_{complete} > 0\right) \\ 1 & \left(n_{complete} = 0\right) \end{cases}$$

where, $n_{complete}$ is the number of successful (task completed) networks in the population, and N is Gaussian random variable.

3.3 Motion Coordination

In the distributed environment, we need a mechanism to determine which agents to be activated. We introduce a simple transfer mechanism to deliver any motivation value and selection probability of agent α_i to be moved is given by

$$Prob(\alpha_i) = \frac{exp(m_i / T)}{\sum_k exp(m_k / T)} \quad (4)$$

where T adjusts randomness of the selection.

An agent will lose one unit of motivation value when it moves one cell width or collides static obstacles and a fixed amount of units when it sends "request-to-

move" signal to mobile obstacles (blocked agents). Instead, by the transfer mechanism, the agent gains a constant motivation value when it receives "request-to-move" signal from neighboring agents and a distributed one from goal attained agent. Goal attained agents will completely lose their currently available motivation value and stay there until they receive motivation value from other agents and are reactivated.

The computation of motivation value for each agent shown in Fig.2 is as follows:

When **E** moves to vacant cell, $m_E = m_E - 1$.

When **C** collides against wall, $m_C = m_C - 1$.

When **B** collides against **D** (**B** is blocked by **D**),

$m_D = m_D + r$ and $m_B = m_B - r$. $(r \geq 1)$

When **B** attains its own goal,

$m_C = m_C + m_B / 3$, $m_D = m_D + m_B / 3$,

$m_E = m_E + m_B / 3$ and $m_B = 0$.

where m_i is the motivation value of agent i and r is the request value to move which gives to the blocked agent.

4 Experiments

We provide some simulation results based on the proposed method. The best result among several trials is shown. Fig.3 shows the test example with six mobile agents. As seen in the figure, without good motion coordination among the agents, deadlock will easily occur. For example, before going to the given goal positions, agents D, E, F are required to make open areas for agents A, B, C. In our experiment, population size is 200, $c_1 = c_4 = 1.0$, $c_2 = 0.3$, $c_3 = 0.1$, $\mu = 0$ and $\sigma^2 = 0$.

To execute the given task in Fig.3, we make the agents learn fundamental motions starting from the simplest problem (Tp-1) in Fig.4. From the result of a preparatory experiment, it turned out that the stochastic selection mechanism is essential to complete the given task and to raise its rate of success though it will not be assured to activate a fixed agent even in the same situation. With respect to the request value to move (r in section 3.3) which is the most important factor in the transfer mechanism of motivation value, we adopted r = 10 and 20 in the following experiments considering the situation of reactivation.

The behavior function of agents is trained step by step using small size problems in Fig.4 by their order before the task starts. Here let n of Ptn-n be the number of problems used for training of the behavior function. So, Ptn-3 means that the task in Fig.3 is executed after training from Tp-1 to Tp-3. Each initial population is generated according to the training condition: Ptn-0 uses only random weights without training any problem, and Ptn-1 takes in only the best individual (i.e. connection weights) obtained during the training of Tp-1 in Fig.4. After trained Tp-2 using the population for Ptn-1, Ptn-2 contains a set of the best individuals for both Tp-1 and Tp-2. Ptn-3 or later is also provided in the same way. By this method, successful tasks are acquired for all except

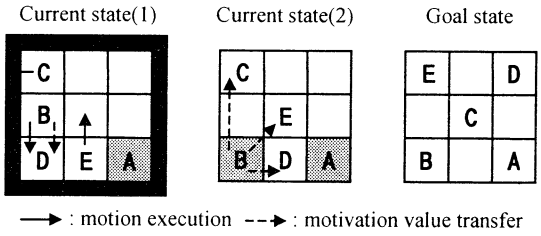


Fig.2 The states for computing motivation value

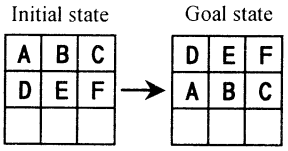


Fig.3 The test example

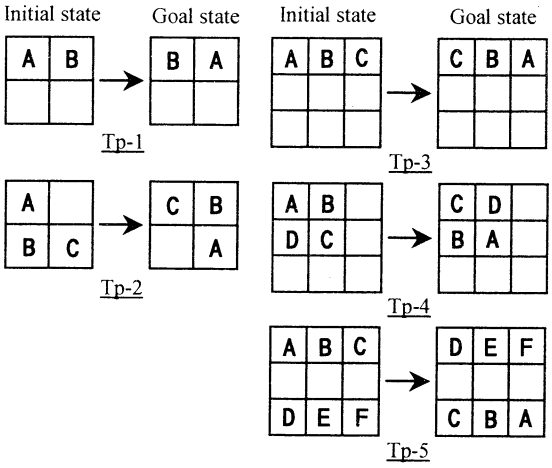


Fig.4 The training set

Ptn-0 and Ptn-1, and then the number of required time step to goal attainment was decreased and the magnitude of the change in distance measure (*DIST* explained above) became smaller as the number of problems used for training increased.

Next, we applied the proposed method to 8-puzzle problem similar to the above task. Fig.5 and Table.1 show particular problem instances in 8-puzzle problem and their characteristics. In Table.1, the error of estimated distance is the value which subtracts the distance measure between the initial state and the goal state from the number of motions for the optimal solution. Combination of the best initial population generated to execute the task in Fig.3 with the best individual obtained during execution of the same task is used as the initial population of networks for solving each instance. Fig.6 shows the change in distance measure during a successful task for each instance, and Fig. 7 illustrates the change of motivation value during the same task in Ex-5. In Fig.6, effective time step is the number of unit motions with actual move and valid time step is that of the motion sequence which removes temporal continuous motions of a certain agent. The magnitude of the change expresses a difficulty of this problem. Agent-A did not move at all. Table.2 shows

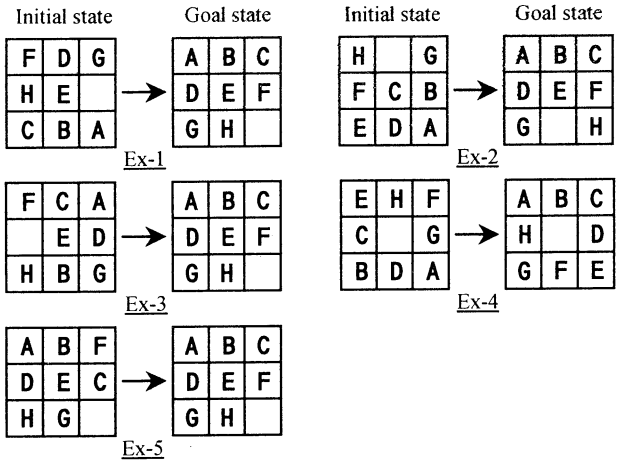


Fig.5 The problem instances

Table.1 The characteristics of the problem instances

	The number of the optimal solution	The number of motions for the optimal solution	The error of estimated distance
Ex-1	40	31(Max)	—
Ex-2	86(Max)	30(Fixed)	—
Ex-3	1(Min)	29(Max)	—
Ex-4	—	24(Max)	0(Min)
Ex-5	—	22(Min)	18(Max)

three kinds of time step and the order of goal attainment for each instance, in which the number 1 and 2 for initial pop mean the network training for r = 10 and 20. From this result, it can be seen that each valid time step is nearly equal to its own optimal time step and its order of goals attains serializable solution except for Ex-5.

The behavior of the agents shown here is essential in the cellular warehouse problem we consider.

5 Conclusions

We proposed a distributed approach to coordinate motions of agents for cellular warehouse problem. Starting from small size problems, and using some example problems which will easily causes deadlocks and reactivation problem, we showed the proposed method is applicable to solve 8-puzzle problem as well as their problems. It was confirmed that a proper exchange of motivation value made a contribution to problem solving. We will investigate how the behavior function is formed as learning results in our future work.

References

[1] Angeline P J (1994), An Evolutionary Algorithms that Constructs Recurrent Neural Networks. IEEE Trans. On Neural Networks, 5-1, 54-65

[2] Minagawa M (1998), Solving Block Stacking Problems in Cellular Space. ANNIE '98, ASME Press, 347-356

[3] Ribeiro F et al (1993), Dynamic Selection of Action Sequences. From Animals to Animats 2, MIT,189-195

[4] Korf R E (1987), Planning as Search: A Quantitative Approach. Artificial Intelligence, 33, 65-88

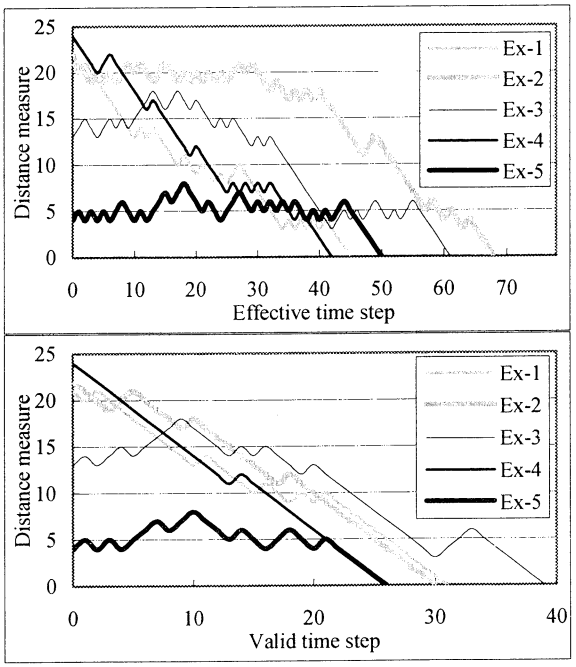


Fig.6 The change in distance measure

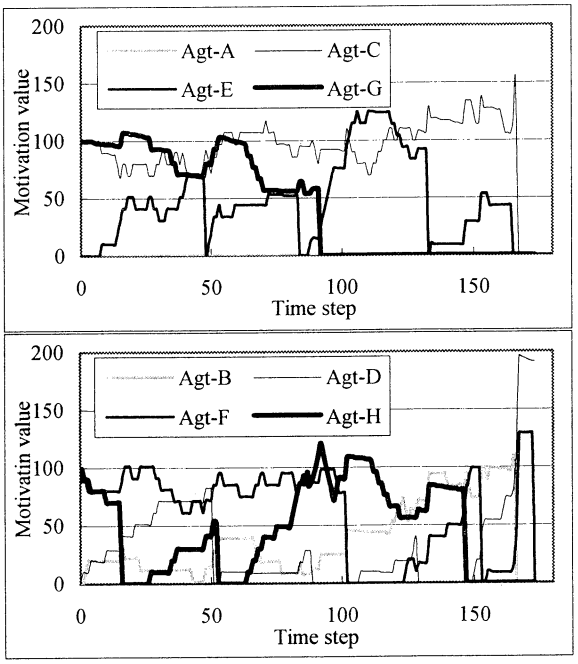


Fig.7 The change of motivation value for Ex-5

Table.2 Three kinds of time step and order of goals

	Request value	Initial pop	Number of time step			Order of goal attainment
			total	effective	valid	
Ex-1	5	1	126	45	31	GDABCHEF
Ex-2	10	1	209	68	30	AGDBCEFH
Ex-3	5	1	225	61	39	GDABCHEF
Ex-4	10	1	127	42	26	CGDEFHAB
Ex-5	10	2	173	50	26	ADGHEBCF

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Integrating Top-down Processing and Bottom-up Processing to Interpret Ambiguous Figures

Yasunori Ohtsuka *, Hiroaki Kudo +, Noboru Ohnishi + and Noboru Sugie*
Department of Electrical and Electronic Engineering
Graduate School of Science and Technology, Meijo University
1-501 Shiogamaguchi, Tempaku-ku, Nagoya, 468-8502, Japan
E-mail address sugie@meijo-u.ac.jp
+Department of Information Engineering
Graduate School of Engineering, Nagoya University

Abstract

It has been well accepted that the human object recognition consists of the interplay of both the bottom-up processing and the top-down processing of viewed images. The perception of ambiguous figures such as Rubin's vase-profile is often discussed in this regard, without having been simulated by computer so far, as illustrations of the interplay, where the focus of attention from time to time is dependent on how the figure of interest is interpreted. That is, depending on the interim interpretation of the figure, the path of visual fixation points (scan-path) will take different courses. The present article is concerned with simulating this kind of phenomenon.

Key words: Visual perception, image understanding, ambiguous figure, Rubin's figure, bottom-up processing, top-down processing.

1 introduction

The human senses information though five senses such as visual sense, auditory sense, and so on. The visual sense is the sense that senses information from the outside. What is considered as one of the roles of the visual system is the recognition of the object.

For thie object recognition, Cavanaugh contended that the bottom-up processing and the top-down processing interact [1].

In this paper we propose a model for interpreting ambiguous figures. The system integrates the bottom-up and top-down processing.

We want implement the mechanism of object recognition. First visual stimuli are analyzed by the bottom-up processing, this is, features are detected and geometrically salient features are fixated one by one and possible semantic interpretations are proposed [2]. Then the top-down processing guides the next fixation point to confirm the proposed interpretation. Depending on whether the confirmation is positive or net the ensuing fixation sequence takes various courses [2].

As the object, we are interested in ambiguous figures, which has at least two interpretations. A preliminary study at cognitive was done by Yamamura et al [5]. As for Necker Cube, a neural implementation was proposed to account for fixation-point interpretation [4]. In this model, the eye is fixated at only one feature.

In this paper we implement a neural implementation of Rubin figure (Fig, 2) recognition, incorporation 'Scan-path.

2 The perception of Rubin figure

2.1 Figure and Ground Issue

In psychology, the concept of figure-and-ground is well known. When we look at the figure shown in Fig.2, the interpretations are either (1) the white region is interpreted as a vase (figure) and the black region as the background (ground), or (2) the black region is interpreted as two face-to-face profiles (figure) and the white region as the background (ground). The two interpretations are mutually exclusive at any moment. The interpretations change from time to time, where the fixation points will play an inportant role as reported by Kawabata et al. [3] as described below.

2.2 The relationship between fixation point and interpretation

In the case where we show the subject Necker Cube as shown in Fig. 1, the probability of two interpreta-

tions varies dependent on where the subject looks at. When the subject fixates at A, plane ABCD tennds to be perceived in front (the numerals associated with the contours denote the probability being about 90

We assume such a dependency will hold in the case of Rubin figure. For example, when the fixation point is at the convex region 0 in the upper-right portion of the white region in Fig.2, we tend to interpret this region either as the upper part of the vase or the head of the profile. The probabilities of two interpretations are not investigated so far.

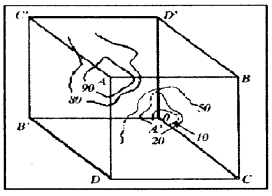


Figure 1: Fixation point dependent interpretation of NeckerCube [2]

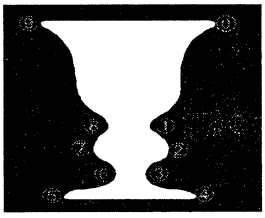


Figure 2: Rubin figure with feature points

Table 1: The contribution ration of each fixation point

the position	the contribution ratio
0	P=1.1
	V=1.5
1	P=1.4
	V=1.2
2	P=1.5
	V=1.1
3	P=1.2
	V=1.4
4	P=1.1
	V=1.5
5	P=1.1
	V=1.5
6	P=1.2
	V=1.4
7	P=1.5
	V=1.1
8	P=1.4
	V=1.2
9	P=1.1
	V=1.5

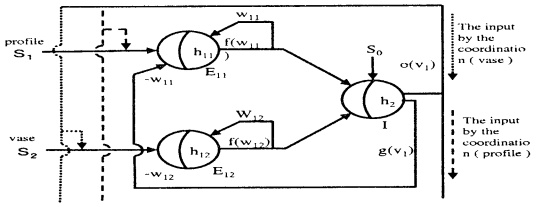


Figure 3: Generalized flip-flop

Tentatively we assume the relative probabilities of two interpretations associated with each feature point in Fig.2 as shown in Table 1 on the basis of intuition. P and V stand for profile and vase, respectively. At the feature point 0, e.g., the relative probabilities of two interpretations, profile and vase, are 1.1 and 1.5,respectively. Thus the interpretation as vase is dominant.

2.3 The neural network model

In the temporal lobe of monkey which is considered as the site of object recognition, Yamane et al. found neurons sensitive to face which respomds vividly to facial parts such as nose, mouth, eye, and so on [5].

In our neural network model, we assume each feature point in Fig.2 is analyzed by a set of neurons. One of the neurons corresponds to the interpretation Profile and the other Vase. The magnitude of the responses designates the strengths of two interpretations. These two neurons E11 and E12 compete with each other to win their own interpretations. Such a winner-take-all network can be realized with a generalized flip-flop proposed by Amari [6]. The network is shown in Fig.3, as the basic competition system.

The competition system is allocated for each feature point and is activated with the fixation, the inputs S1 and S2 being the values shown in Table 1. By fixating feature points one by one, the coorperation among the neurons with the same interpretation, in the interconnected competition systems, takes place. Eventually the whole system may get stable with the dominance of either of two interpretations as a result of winner-take-all characteristics.

2.4 Scan-path

According to the Scan-path hypothesis due to Noton and Stark [2], the locus of fixations (Scan-path) at an object is specific to the object and probably to the observer. That is, each object is associated with its own scan-path, which may be considered as the model of object. The role of the scan-path is to test

the hypothesized interpretation of object of concern. If the hypothesis turns out incorrect, another interpretation is tested with another specific scan-path. In Fig.4 is shown a picture of a face and the locus of fixation points of a subject. The data acquired through 4 trails are shown.

In our model the above scan-path scheme is embedded. We assume the scan-paths for two interpretations, profile and vase. They are shown in Fig.5. Depending on the interpretation, the importance of each feature will be different semantically, which is reflected in the difference of the scan-paths. Thus in the model, the scan-path determines the sequence of fixations. This process constitute the top-down processing.

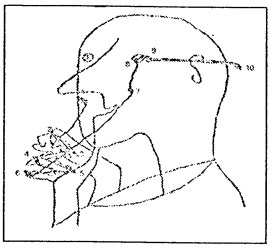


Figure 4: Visual stimulus and the scan-path[2]

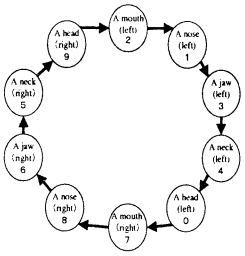


Figure 5: The scan-path for the profile

3 The flow of the processing

The overall flow of the processing is shown in Fig.6 which consists of two main components, bottom-up and top-down components.

3.1 The bottom up processing

A) The retina
The stimulus figure is projected on the retina. The image is processed while it is transmitted from the retina and visual areas in the cerebrum to extract geometrically salient features such as sharp corners of edges, end points of edges, and so on. These features are combined to form such parts of objects as nose and eyes. The finally objects such as faces are recognized. In our model the retina component can extract feature points 0 ~9 in Fig.2 following the extraction of contours.

B) Competition
As described in 2.3, once a feature point is fixated, the corresponding generalized flip-flop is supplied with inputs. Then competition between neuron E11 and E12 takes place.

3.2 The top down processing

A') Coordination
Through the competition, the winner of the interpretation will be determined. Note, however, the cooperation among interconnected neurons of the same interpretation constitutes another important interaction. This process is of top-down nature.

B') Object model
The scan-path is used to guide the eye movement and is considered as the model of object. The model guides active vision.

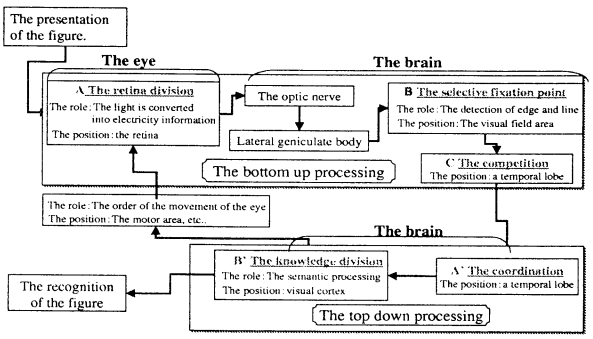


Figure 6: The flow of the processing

4 The simulation

4.1 The description of the display screen

With the idea of the processing described in 3 chapters, the program was made, and the simulation was tried.
Figure 7(a) is display screen in the process stage.
The left side in the screen shows which graphic region is noticed, and the center screen shows in which region the profile was understood. The right side screen shows in which region we understood it the vase.

4.2 Simulation result

A) The case in which we recognize as a profile
Fig. 7 (a) through (f) show the time course of in-

terpretation of Rubin figure. Some regions were interpreted vase at first. However, through competition and cooperation processes, the final interpretation of the whole figure was profile.

B) The case in which we recognize as a vase

Similarly Fig. 8(a) through (f) show the time course of interpretation of Rubin figure. Some regions were interpreted profile at first. However, through competition and cooperation processes, the final interpretation of the whole figure was vase.

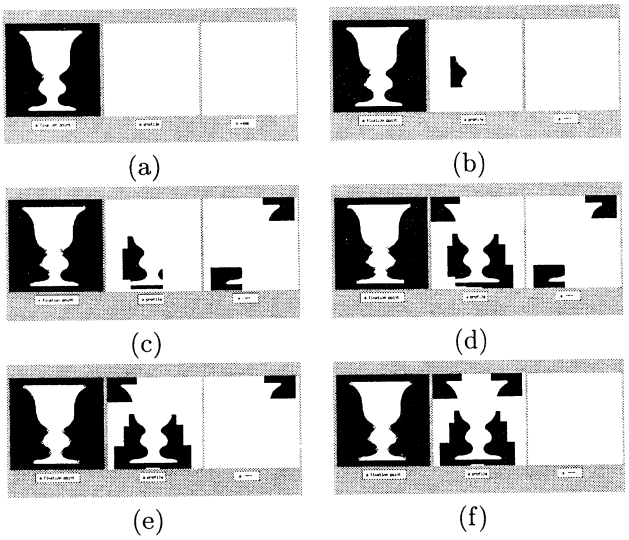


Figure 7: The time course of arriving at the interpretation as profile

5 Conclusion

We implemented a neural network model which interprets Rubin figure by integrating the bottom-up processing and the top-down processing. The network consists of interconnection of generalized flip-flops. Within a flip-flop, winner-take-all type competition takes place. Based on the most probable interpretation of fixated features, the top-down processing determines the next fixation point to test the tentative interpretation. The new fixation activates the corresponding generalized flip-flops. This process is based on the scan-path theory [2]. Through the interconnection of generalized flip-flops, a cooperative process is realized. As a whole the network will proceed to a single interpretation. The time courses of arriving

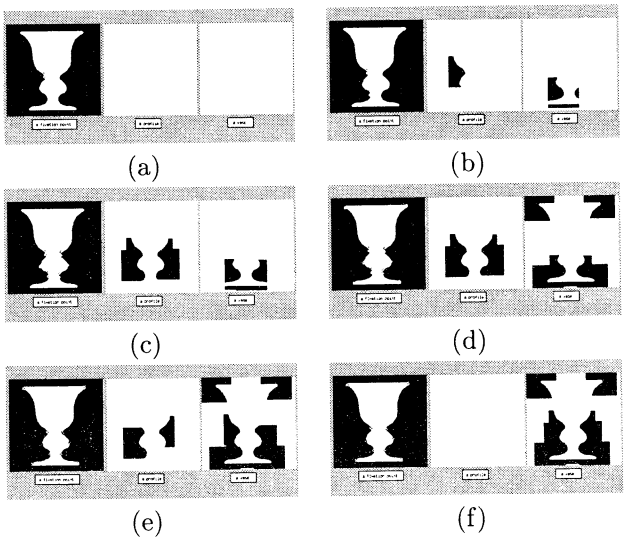


Figure 8: The time course of arriving at the interpretation as vase

at the interpretations of profile and vase were successfully simulated. The future issues are (1) the contribution ratio of the interpretation at each fixation point should be determined by experiments; (2) the modeling of objects should be pursued further to include issues related to aspect, size,and so on.

References

[1] P.Cavanagh, "Top-Down Procesing in Vision," *The MIT Encyclopedia of the Cognitive Sciences*, pp. 844-845,1999.

[2] D.Noton and L.Stark, "Eye Movements and Visual Perception," *Scientific American*, 224, pp. 35-43, June, 1971.

[3] N.Kawabata, K.Yamagami, and M,Noaki, "Attention and Depth Perception," *Perception*, 15,pp. 563 − 572,1986.

[4] T Yamamura, M Ohora, N Ohnishi et al., "Simulation for an Ambiguous Figure," *SICE Tech.Reprt*, 31, 8, pp. 1242-1244, 1995. (in Japanese)

[5] S, Yamane et al., "What facial features activate face neurons in the inferotemporal cortex of the monkey?" *Exp. Brain Res.*, 73, pp. 209-214, 1988.

[6] S, Amari., "Mathematical Principles of Neural Networks." *Tokyo.:Sangyo Tosho*, pp. 205-214, 1978. (in Japanese).

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

DETECTING EYES USING CIRCULAR HOUGH TRANSFORM AND TEMPLATE MATCHING

Mohamed Rizon and Tsuyoshi Kawaguchi
Department of Computer Science and Intelligent Systems,
Oita University, Oita, 870-1192 Japan
E-mail: {mohamed,kawaguti}@csis.oita-u.ac.jp

Abstract: *In this paper we propose a new algorithm to detect the pupils of both eyes from a human face in an intensity image. First, feature points which are the candidates for the pupils of both eyes are extracted from the face image by using the feature template proposed by Lin and Wu. Next, the proposed algorithm computes a cost for each pair of feature points satisfying a spatial constraint. The cost is computed by searching for a circular region corresponding to the iris around each feature point. Finally, the algorithm determines a pair of feature points with the smallest cost to be the pupils of both eyes. As the result of the experiment using all faces without spectacles in the face database of the University of Bern, the success rate of the proposed algorithm was 95.3[%] on the average. And, if looking-down faces are excluded, the success rate of the proposed algorithm was 98.7[%] on the average.*

Keywords: *Face recognition, face detection, facial features.*

I. INTRODUCTION

The face recognition has many applications such as personal identification, criminal investigation, security work and login authentication. Automatic recognition of human faces by computer has been approached in two ways: holistic and analytic. The holistic approach treats a face as 2D pattern of intensity variation. The analytic approach recognizes a face using the geometrical measurements taken among facial features, such as eyes and mouth.

In the analytic approach, reliable detection of facial features is fundamental to the success of the overall system. In addition, it is also important in the holistic approach for the face normalization. The face in the input image must be normalized to a standard size, location and orientation before it is matched to database faces. And, such normalization can be done using some geometrical measurements among facial features.

The eyes can be considered salient and relatively stable features on the face in comparison with other facial features. And, if both eyes are detected, the positions of other facial features can be estimated using the positions of both eyes [1]. In addition, if both eyes on the image face can be detected, the face can be easily normalized using geometrical measurements between them [1].

Eye detection also plays an important role in applications such as video conferencing and vision assisted user interfaces.

Thus, many eye detection algorithms have been proposed. Most of the eye detection algorithms previously reported are classified into three types. The first approach [1] locates eyes using cross-correlation between the eye template and the image. The second approach [2], [3] uses the principal component analysis to locate eyes. And, the third approach [4] uses deformable templates to detect eyes.

However, the first and the second approaches require the normalization of the image face in its size and orientation when the variations in size and orientation of the image face are not small. And, the third approach has a drawback that the initial position of the eye template has to be set approximately to the position of the eyes.

Recently, Lin and Wu [5] proposed a novel facial feature detection algorithm. The algorithm computes a cost for each pixel inside the face region using a generic feature template and selects pixels with the largest costs as facial feature points.

We verified by our experiments that it is difficult to detect both eyes by only using the algorithm of [5] because pixels with the largest costs are not always facial feature points. But, the algorithm is very attractive because the algorithm can detect the candidates for facial features even when size and orientation of the image face are unknown and even when some regions of the face exhibit low contrast.

In this paper we propose a new algorithm to detect the pupils of both eyes. The proposed algorithm uses the algorithm of [5] to detect the candidates for the pupils of both eyes.

II. ALGORITHM OF LIN AND WU TO DETECT FACIAL FEATURES

We first give a brief description of the algorithm proposed by Lin and Wu [5] to detect facial features.

The algorithm extracts the face region from the input image using a region-growing method. Next, for each pixel (x,y) inside the face region, the algorithm computes a cost C(x, y) by

$$C(x,y) = C_1(x,y) + C_2(x,y) \tag{1}$$

Let S(x,y) denote the square region with center (x,y) and side-length d. Then, $C_1(x,y)$ is given by

$$C_1(x,y) = \sum_{j=y-d/2}^{y+d/2} V_r(j) + \sum_{i=x-d/2}^{x+d/2} V_c(i) \tag{2}$$

$V_r(j)$ and $V_c(i)$ are the mean crossing numbers of row j and column i which are defined as follows.

Let I(i,j) denote the intensity values of pixels (i,j) in the image. Then, for each row j, $V_r(j)$ represents the number of pixels (i,j), $x-d/2 \leqq i \leqq x+d/2$, such that one of the I(i−1,j) and I(i,j) is greater than μ plus K and the other is smaller than μ minus K where μ is the average intensity of pixels (i,j), $x-d/2 \leqq i \leqq x+d/2$ and K is a constant. $V_c(i)$ denotes the number of pixels (i,j), $y-d/2 \leqq j \leqq y+d/2$, such that one of the I(i,j−1) and I(i,j) is greater than μ plus K and the other is smaller than μ minus K.

$C_2(x,y)$ is a function which evaluates the intensity difference between the central part and the boundary parts of S(x,y).

After partitioning the face region into the left eye subregion, the right eye subregion and the mouth subregion, the algorithm selects three pixels with the largest costs, one from each subregion, and determines these pixels to be the positions of the left eye, the right eye and the mouth.

The feature template used in the algorithm of [5] is very attractive because it is robust even when some subregions of the face exhibit low contrast. But, we verified by our experiments that although the costs of facial feature points are relatively large, the probability that a facial feature point has the largest cost inside each subregion is low. Thus, the eye detection algorithm proposed in this paper uses the algorithm of [5] to detect candidates for both eyes.

III. THE PROPOSED ALGORITHM TO DETECT BOTH EYES

In this paper we assume that the face images are head-shoulder images with plain backgrounds. The proposed algorithm first extracts the face region from a head-shoulder image using the vertical and the horizontal projection values computed on the edge image. Next, the algorithm extracts valley regions from the face region and then detects feature points of the face using the algorithm of [5]. (In this paper, feature points mean the candidates for the pupil positions of both eyes.) After that, using a procedure proposed in this paper, the algorithm computes a cost for each pair of the feature points satisfying a spatial constraint. Finally, the algorithm determines a pair of feature points with the smallest cost to be the pupils of both eyes.

Fig. 1 shows the feature points extracted by the feature detection procedure and Fig.2 gives the positions of the pupils detected by the proposed algorithm.

A. Costs of Feature Points

Let $B_i = (x_i,y_i)$ denote feature points obtained by the feature detection procedure. First, the proposed algorithm places the template of Fig.3 at each feature point B_i and measures the separability between two regions R_1 and R_2 in the template using the separability filter proposed by Fukui et al [3]. Let $\eta(r)$ denote the separability between two regions R_1 and R_2 in the template with size r. Changing the size r in the range of $\{r_L, r_L+1, \ldots, r_U\}$, we find the size r maximizing $\eta(r)$. And, let r_i denote the size r maximizing $\eta(r)$. We call r_i

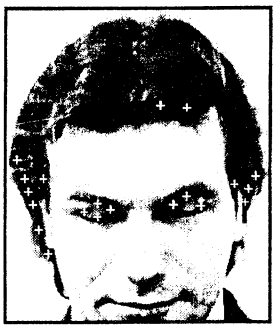


Fig. 1 The feature points extracted by the feature detection procedure

Fig.2 The positions of the pupils detected by the proposed algorithm

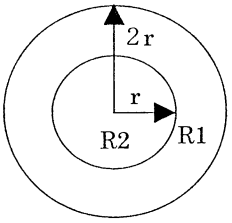


Fig.3 An eye template

the size of the iris at B_i. In the experiments, r_L and r_U were set to 5 and 7 [pixels], respectively.

Next, the proposed algorithm measures the separabilities $\eta_{23}(i)$, $\eta_{24}(i)$, $\eta_{25}(i)$ and $\eta_{26}(i)$ using the templates of Fig.3 with sizes r_i, where $\eta_{kl}(i)$ denotes the separability between regions R_k and R_l.

Thirdly, the proposed algorithm applies the Canny edge detector [6] to the face region and then detects a circle around B_i using the circular Hough transform [7]. We give the equation of a circle as follows.

$$(x-a)^2+(y-b)^2=r^2$$

where (a, b) is the circle center and r is the radius. Let (a,b,r) denote the circle obtained by the circular Hough transform and V(a,b,r) denote the vote for (a,b,r). Then, we call V(a,b,r) the vote for B_i.

Finally, the proposed algorithm gives the cost of each feature point B_i by

$$C(i)=C_1(i)+C_2(i)+C_3(i)+C_4(i) \quad (1)$$

$$C_1(i)=\frac{|\eta_{23}(i)-\eta_{24}(i)|}{\eta_{23}(i)+\eta_{24}(i)},\quad C_2(i)=\frac{|\eta_{25}(i)-\eta_{26}(i)|}{\eta_{25}(i)+\eta_{26}(i)}$$

$$C_3(i)=\frac{V_{max}}{V(i)},\quad C_4(i)=\frac{U(i)}{U_{av}}$$

where

$V(i)$: the vote for B_i given by Hough transform,

V_{max} : the maximum of V(i) over all feature points,

$U(i)$: the average intensity inside R_2,

U_{av} : the average of U(i) over all feature points.

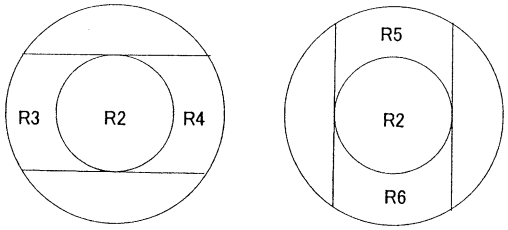


Fig.4 The templates used to compute $C_1(i)$ and $C_2(i)$

B. Select a Pair of Feature Points Corresponding to the Pupils of Both Eyes

For each pair of feature points B_i and B_j, let d_{ij} and θ_{ij} denote the length and the orientation of the line connecting B_i and B_j. The proposed algorithm computes a cost for each pair of feature points B_i and B_j such that $L/4 \leqq d_{ij} \leqq L$ and $-30^\circ \leqq \theta_{ij} \leqq 30^\circ$ where L is the difference between the x-positions of the left and right contours of the face. And, a pair of feature points with the smallest cost is determined to be the pupils of both eyes.

The cost of a pair of feature points B_i and B_j is given by

$$F(i, j) = C(i) + C(j) + 1/R(i, j) \quad (2)$$

where C(i) and C(j) are the costs computed by Eq.(1) and R(i, j) is the normalized cross-correlation value computed by using an eye template, which is produced by manually cutting off the eye region from a face image(see Fig. 5).

For each pair of feature points B_i and B_j, the algorithm computes R(i, j) by using the following procedure.

(1) Normalize the scale, translation and orientation of the eye template so that the positions of the pupils in the template are placed at B_i and B_j.

(2) Compute the normalized cross-correlation value R(i,j) between the template and the image.

(3) If R(i ,j) is smaller than 0.1, then set R(i, j) to 0.1.

IV. EXPERIMENTAL RESULTS

We made experiments to evaluate the performance of the proposed algorithm using face images taken from the face database of University of Bern, Switzerland. This database has 150 faces with spectacles and 150 faces without spectacles. The sizes of the images are 512×342.

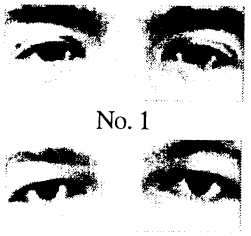
We made experiments using all faces without spectacles, which consists of 10 views of 15 persons. The ten views include faces looking to the right, to the left, downward and upward.

The performance of the proposed algorithm slightly depends on what template is used to compute the cross-correlation value R(i,j). Therefore, we made experiments using two templates, which are shown in Fig.5. The results of the experiments are given in Table1. Examples of the images for which the proposed algorithm could correctly detect the pupils of both eyes are shown in Fig. 6. And, Fig. 7 shows examples of the images for which the proposed algorithm failed the correct detection of the pupils.

The time needed by the proposed algorithm to detect the pupils of both eyes was about 3.6 [seconds] on the average of 150 images by a computer whose CPU is Pentium III, 800 MHz.

To evaluate the usefulness of the proposed algorithm, we compared the performance of the proposed algorithm with the performances of the template matching and the eigenspace method. The algorithm which we call template matching determines a pair of blobs B_i and B_j with the maximum cross-correlation R(i,j) to be the irises of both eyes. The algorithm which we call the eigenspace method selects a pair of blob corresponding to the irises of both eyes by using the algorithm of [2]. The algorithm is the same as the algorithm of [3], except the function used to compute the matching errors of blobs. We used the error function shown in [2].

Table 2 shows the results of the experiments.



No. 1

No.2

Fig.5 The templates used for the computation of R(i, j)

V. CONCLUSIONS

In this paper we proposed a new algorithm to extract the pupils of both eyes from a face image.

Most of the eye detection algorithms previously reported use the template matching or the principal component analysis. However, these algorithms require the normalization of the image face in its size and orientation when the variations in size and orientation of the image face are not small. On the other hand, the eye detection algorithm proposed in this paper need not normalize the size and the orientation of the face in the image. In addition, although the proposed algorithm uses an eye template cut off from a face image, the performance of the proposed algorithm is not sensitive to the variation of the templates. We made experiments using all faces without spectacles in the face database of the University of Bern. As the result, the success rate of the proposed algorithm was 95.3[%] on the average. And, if looking-down faces are excluded, the success rate of the proposed algorithm was 98.7[%] on the average.

ACKNOWLEDGEMENT

This work is partly supported by the Grant-in-Aid for Scientific Research under Grant No. 11680396, by the Japan Society for the Promotion of Science. And, the first author is supported by the Hitachi Scholarship Foundation.

REFERENCES

[1] R. Brunelli and T. Poggio, "Face recognition: features versus templates," IEEE Trans. PAMI, Vol.15, No.10, pp.1042-1052(1993)

[2] B. Moghaddam and A. Pentland, "Probabilistic visual learning for object representation," IEEE Trans. PAMI, Vol. 19, No. 7, pp.696-710(1997)

[3] K. Fukui and O. Yamaguchi, "Facial feature point extraction method based on combination of shape extraction and pattern matching," IEICE Trans., Vol.J80-D-II, No.8, pp.2170-2177(1997)

[4] A. L. Yuille, P. W. Hallinan and D. S. Cohen, "Feature extraction from faces using deformable templates," Int. J. Computer Vision, Vol.8, No.2, pp.99-111(1992)

[5] C.H. Lin and J.L. Wu, "Automatic facial feature extraction by genetic algorithms," IEEE Trans. Image Processing, Vol.8, No.6, pp. 834-845(1999)

[6] J. Canny, "A computational approach to edge detection," IEEE Trans. PAMI, Vol.8, No.6, pp.679-698(1986)

[7] C. Kimme, D. Ballard and J. Sklansky, "Finding circles by an array of accumulators," Commun. of ACM, Vol. 18, No. 2, pp.120-122(1975)

Table1 The results of the experiments using 150 faces without spectacles taken from the face database of University of Bern.

Template	The number of images for which the proposed algorithm could correctly detect the irises of both eyes	
	N1	N2
No.1	142	118
No.2	144	119

N1 is the result of the experiments using all of the 150 faces and N2 is the result of the experiment using 120 faces in which the looking-down faces are excluded.

Table2 The comparison of the performances of the proposed algorithm, the template matching and the eigenspace method

Algorithm	Success Rate [%]	CPU Time [seconds]
Proposed Algorithm	95.3*	3.6
Template Matching	81.2*	1.5
Eigenspace Method (50 training samples)	90.7** 86.0+	5.2
Eigenspace Method (100 training samples)	92.7** 80.0+	5.4

* The average of the results obtained by using two templates shown in Fig. 5.

** The result for the 150 test images which include all images used as training samples.

+ The result for the test images that were not used as training samples.

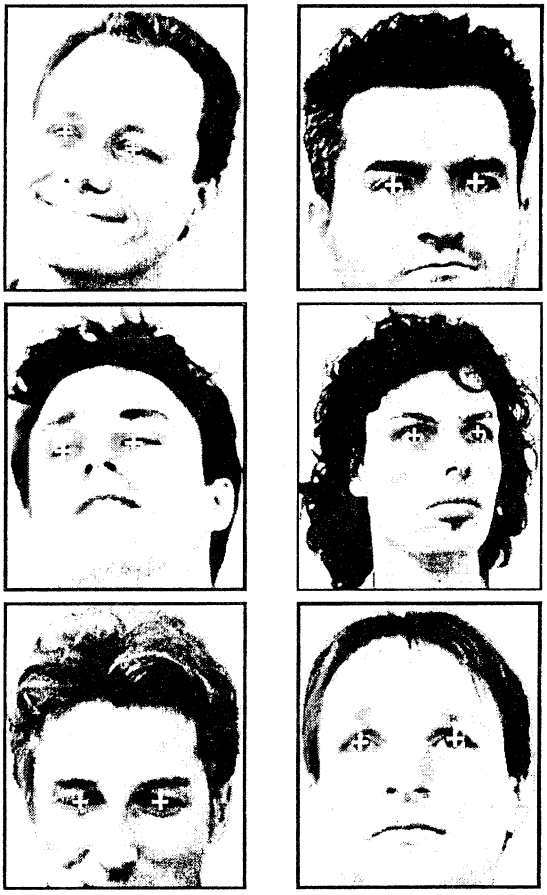


Fig.6 Examples of the images for which the proposed algorithm could correctly detect the pupils of both eyes.

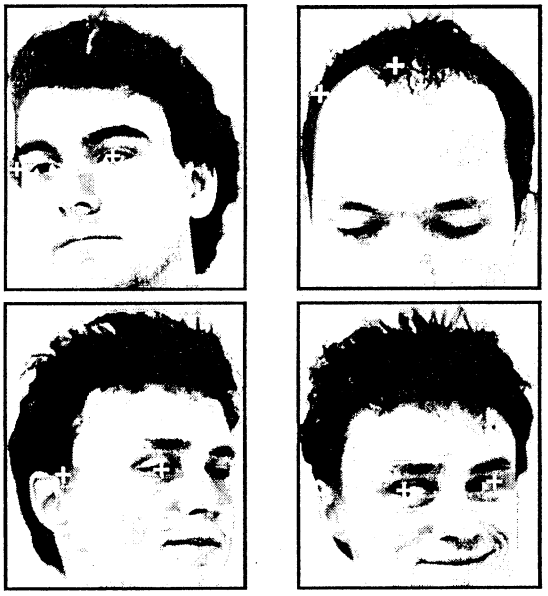


Fig.7 Examples of the images for which the proposed algorithm failed the correct detection of the pupils.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A COSIDRERATION ON PHOTOVOLTAIC POWER GENERATION SYSTEMS

Masanori Sugisaka , Kiyokazu Nakanishi and Noriaki Mitsuo
Department of Electrical and Elctronic Engineering
Faculty of Engineering, Oita University
700 Dannnoharu, Oita City , 870-1192 , Japan
Tel: +81-97-554-7831
Fax: +81-97-554-7841
E-mail: { msugi , nakanisi , nmitsuo } @cc.oita-u.ac.jp

Abstract

In our laboratory, the control aspects are investigated in the photovoltaic power generation systems (PV systems).

The PV system is very good for earth environment, but if it connects to power network system, many problems are raised (protection, voltage, harmonics etc.).

In this paper, we present the result of the basic studies for the building of the PV system that amplifies the electric energy obtained from the solar cell.

We consider electronic circuits in order to protect the PV system from power surge induced by lightning and also design an electronic circuit in order to detect defaults in the power network system.

We would like to integrate these circuits into the PV system by considering its control equipment build by 8-bit microcomputer using various control theory (fuzzy, neural network etc.).

1. Introduction

In the 20th century, our industry and living standard were developed remarkably based on fossil fuel such as petroleum and coal. However, there is not much fossil oil.

When dusts burn they have caused various pollution problems. The avoidance from fossil fuel is future theme because electric power is more consumed in recent years[1].

PV system is thought as one of the way of power generation that is clean and gentle for global environment, but it is actual states that PV system ought not to spread, in spite of some countries' government carry out policy for spreading the PV systems.

Therefore, in Japan, it is in the present situation that the spread percentage in the home of the PV system is stagnating[2].

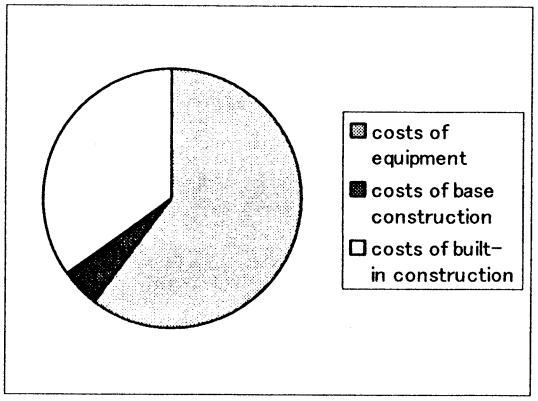


Fig.1 Costs of Installing PV System in Japan[3]

We consider that high cost of the system impedes progress. If the cost saving is available, the PV system can spread more and more.

In this paper, we consider whether power conditioner that is nucleus of PV system can be improved low cost and high performance.

To be realized, we would like to reexamine construction of system from control engineering aspects and moreover construct a new system. Therefore, we don't consider aspects of properties of matter like structuring solar cells.

2. Structure of PV system

At present, PV system put in the market is structured:

- Solar cells (Panel)
- Power conditioner
- Battery

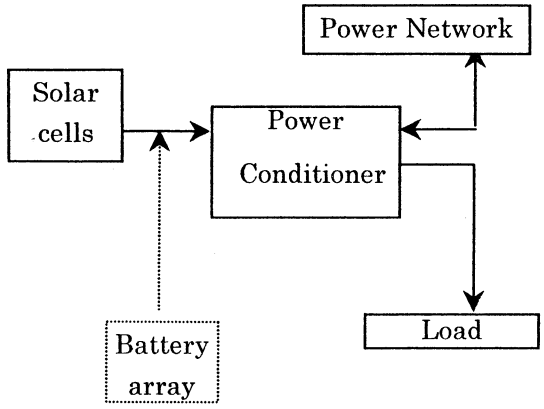


Fig.2 Typical figure of PV system[3][4]

Power conditioner plays an important role. Power conditioner includes :

(1) DC-AC converter,
(2) controller for voltage and current,
(3) controller for maximum power point tracking,
(4) safety device,
(5) and functions for protecting PV system from surge induced from power networks.

In it, some of the most exciting advances in the solar industry have been recent changes in inverter (DC-AC converter) technology. Inverters are a vital component of most solar electric systems, converting direct current energy at a battery bank to alternating current that can be used by standard appliances.

And a device to be remarked is a second diverse controller. Controllers carried important part. Some controllers are needed to bring out power from solar cells.

Other controllers play some of important roles for connecting power network. One of the roles is detecting various faults and disaster in power network. For example, brake down power network, induced lightning stroke,etc.. If any accident occurs, not only in the PV systems equipments are damaged, but also during repair work, the workers get an electric shock because of power from PV system.

We would like to consider reforming whole structure of power conditioner included inverter and various controllers.

3. Our survey of PV system

It is the step to search the parameter to carry forward the verification of the device for the electric power that is in the market at present and to draw out the ability of the device sufficiently.

We would like to consider improving PV system from the following viewpoint.

(1) Examine the devices of characteristic that is used for power electronics, and investigate each advantage and problem point.

(2) Investigate general Power Operational Amplifier and it adopts which suited PV system.

(3) Reconstruct faults and disasters detection system, especially lightning stroke and cut down power network.

(4) Reconstruct methods of control based-microcomputer-using technology of learning theory (neural theory, fuzzy theory, etc.)

4 . Pre-experimentation

We did the following as the basis experiment in case of this study.

It experimented in the voltage amplification using operational amplifier and IGBT(Insulated Gate Bipolar Transistor) using for current amplifier.

In our experimentation, we use battery array as imaginary solar cells. And as operational amplifier, using PA88 made by APEXmicrotechnology co.. These devices have durability for large voltage [5][6]. So, using as current amplifier, 2MBI50N-060 made by Fuji Elec.

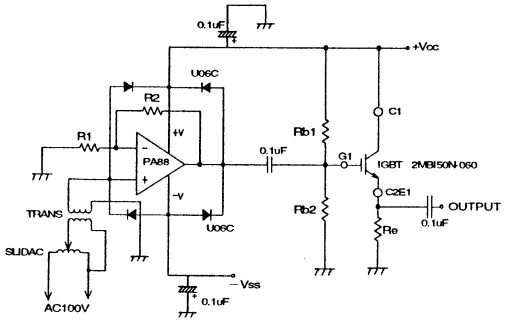


Fig.3 Circuit of this experiment

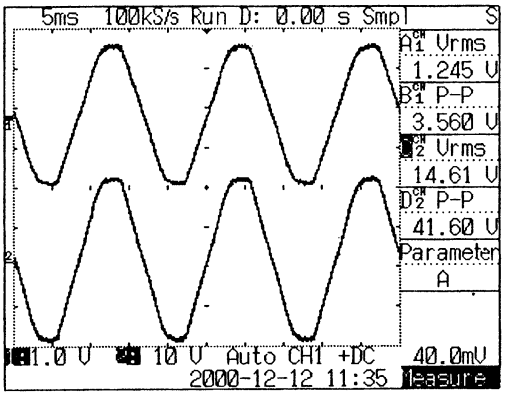


Fig.4 Waveform of output of PA88

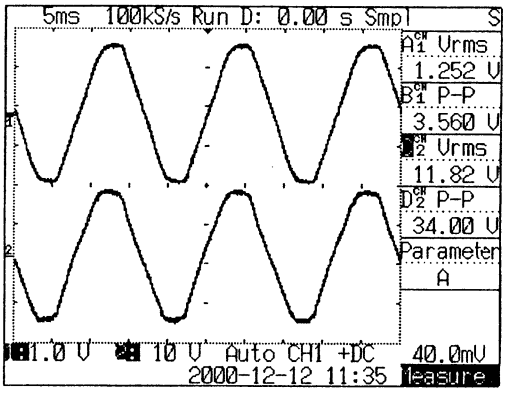


Fig.5 Waveform of output of IGBT

In waveform figures, upper wave is input voltage and lower wave is output voltage.

Moreover, condition of this experimentation is as follows:

Table1. Condition of this experimentation

Input AC voltage	1.25 [V]
Vcc (DC power)	±25.5[V]
R1	33[kΩ]
R2	330[kΩ]
Gain of OP Amp	10
Load	Non-Load

By this experiment, voltage drop exists some direct voltage, but that is gotten from solar cells can be output in the alternating current with operational amplifier. In other word, we can consider that possibility exists power operational amplifier as DC-AC converter (inverter).

5 . Conclusion

In this experimentation, we convinced that operational amplifier and IGBT could use DC-AC power converter.

Through this result, circuit composition can be simply, we consider that can contribute to the cost reduction of power conditioner.

But, we haven't experimented practical load test, yet.

From now on, we'll development current amplifier suited present voltage amplifier. and then It thinks that it wants to suggest a study that it is possible to attempt cost down without lowering the efficiency of power conditioner.

Moreover, we would like to construct safety system that

6. References

[1]M.Sugisaka and K.Sakamoto "The development for inverter for generation of electricity form solar panel" Proc. of the fifth Int. on Artificial Life and Robotics, vol.2, pp621-624, Jun. 2000

[2] key center for photovoltaic engineering "Annual 1999 report" The University of New south Wales, pp4, 1999

[3]NEDO(New Energy and Industrial Technology Development Organization) "The trend of project new energy project real state investigations" http://www.nedo.go.jp/www/intro/taiyo/5.html,1996

[4]K.Kurokawa and S.Wakamatsu " Photovoltaic Power generation system design guidebook" Ohm sya pp24 , Aug.1994

[5] Japan Photovoltaic Energy Association " Design and construction of Photovoltaic Power generation system" Ohm sya pp38-49, Feb. 2000

[6]M.Suzuki " sequel design of transistor circuit " CQ publishing pp 108-117,134-137, 1992

[7]Apex microtechnology " Vol.8.7 Databook CD-ROM " pp329-332

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

A Robot Location Solution in A Certain Circumstance

Zhuanxin Li Jing Pei Huaiyu Wu
Optical Memory National Engineering Research Center
Tsinghua University, Beijing 100084, P.R. China
E-mail: lizx@post.pim.tsinghua.edu.cn

Abstract

An automatic robot was constructed by the Development Team of Tsinghua University for ROBOCON2000. In order to locate the robot for other controls, an approach about this task was processed from Oct. 1999 to Feb. 2000. Although there are several classic methods to locate robots in a certain situation, such as sonar detection, infrared ray measurement and so on, these traditional methods seem not suitable in the contest environment, thus a new concept was introduced with a usual means. This paper aims to present the special background and to describe the solution in the practice.

Key words: *Robot, Navigation, Location, Algorithm*

1. Background

The automatic robot was constructed for ROBOCON2000, a robot contest held by NHK Company in 2000. According to the rules of ROBOCON2000, the automatic should move to a destination where 100 balls are located, after taking a number of balls, then navigate to a comfortable position and use them to shoot objectives. The more objectives are shot, the more mark the team will obtain.

White cross-line is drawn for guidance. So one direct way to navigate the automatic robot is that let it detect the white line and follow them in a definite model. It seems available, but you should remember that it's a contest. If you were slower, your adversary would snatch most of the balls and have more chance to get high mark. Therefore in such a competitive situation, moving along with white line is not a wise choice.

In many kinds of circumstances, utilizing sonar or infrared to measure distance is a useful means. However, the wall around the field for the contest is only 30 centimeters high, and there are a number of balls on the ground, which height are about 8 centimeters. Hence, the sonar or infrared signals can just work in a narrow range. At the same time the noise and shinning light in the hall can also affect the signals. And the data received from the sonar or infrared signals are not available for the uncertainty of status angle of the automatic robot. This paper presents a new concept solution which is practicable to the competitive situation

2. The solution

In a plain ground, a two-dimension coordination system could be set when choosing an original point. Since the position of every objective is fixed, everyone has its own coordination in the system. If we adopt some means to acquire the coordination of the automatic robot, it can make decision according to the coordinations of the objectives and itself. The physical model is shown in Fig. 1.

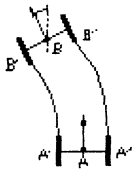


Fig. 1

When the automatic robot moves from Point A to Point B, it turns over an angle θ. The channel of the left wheel could be a random arc linking Point A′ and Point B′, while the one of the right wheel is an arc with L-length from the arc A′B′.

If the frequency to get signals from encoders is high enough, arc AB can be seemed as a line (Fig. 2). Nonetheless, it is more suitable to use an arc as model, since in most conditions the left wheel can be synchronous with the right one.

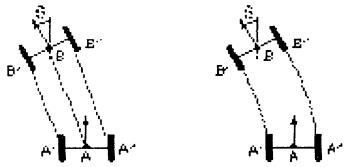


Fig. 2

In order to get an algorithm for programming, a mathematic model should be established for computation. It can be divided into four sub-models, such as turning left while processing forward, turning right while moving backward, and so on. Following is illustrated for locomoting forward. (Fig. 3)

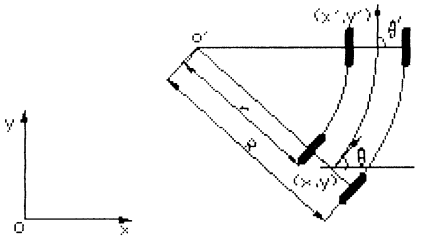


Fig. 3

Set coordination xOy. The initial direction angle of the automatic robot is θ, while the ending direction angle is θ'. Some other parameters that can be known are x, y, D, Sl, Sr, and we shall obtain x', y', θ'.

(x, y): the initial coordination of the robot;

D: the length between two wheels;

Sl: the length of arc that left wheel passes;

Sr: the length of arc that right wheel passes;

(x′, y′): the ending coordination of the robot.

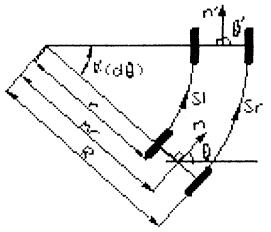


Fig. 4

According the geometrical definition (Fig. 5), the turning angle between vector n and n' is α, because each of them is vertical with one side of angle α. So we can obtain following results.

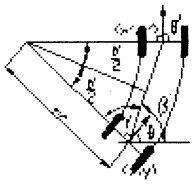


Fig. 5

$$\alpha = \frac{Sl}{r} = \frac{Sr}{R} = \frac{Sr - Sl}{R - r} = \frac{Sr - Sl}{D} \tag{1}$$

$$d\theta = \frac{Sr - Sl}{D} \tag{2}$$

Obviously we can get the following relation (3) from (1) and (2).

$$\begin{cases} d\theta = \dfrac{Sr - Sl}{D} \\ \theta' = \theta + d\theta \end{cases} \tag{3}$$

In addition, mr can be described as (4).

$$mr = \frac{1}{2}(r + R) = \frac{1}{2}\left(\frac{Sl}{\alpha} + \frac{Sr}{\alpha}\right) = \frac{D}{2} \cdot \frac{Sr + Sl}{Sr - Sl} \tag{4}$$

According the relationship between angles, there is formula (5)

$$\begin{cases} \gamma = \dfrac{\pi - \alpha}{2} \\ \beta = \theta + \left(\dfrac{\pi}{2} - \gamma\right) \end{cases} \tag{5}$$

Therefore,

$$\beta = \theta + \frac{\alpha}{2} \tag{6}$$

As shown in Fig. 6, L can be denoted as (7).

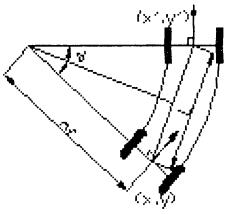


Fig. 6

$$L = 2mr \cdot \sin\left(\frac{\alpha}{2}\right) \tag{7}$$

Thus, the increments of x and y can be given by (8)

$$\begin{cases} dx = L \cdot \cos\beta = 2mr \cdot \sin\left(\dfrac{\alpha}{2}\right)\cos\left(\theta + \dfrac{\alpha}{2}\right) \\ dy = L \cdot \sin\beta = 2mr \cdot \sin\left(\dfrac{\alpha}{2}\right)\sin\left(\theta + \dfrac{\alpha}{2}\right) \end{cases} \tag{8}$$

3. Conclusions

As the discussion above, the general location algorithm can be drawn.

If the definitions are given as follows,

1. Denotation of Sl or Sr is positive when the automatic robot moves forward, vice versa;
2. The angle θ is defined as the range of $-180° < \theta < 180°$, and is positive as anti-clockwise, negative as clockwise;

the results from the four sub-models can be uniformed

into one format below, which is very comfortable to locate the automatic robot.

$$\begin{cases} d\theta = \dfrac{Sr - Sl}{D} \\ mr = \dfrac{D}{2} \cdot \dfrac{Sr + Sl}{Sr - Sl} \\ dx = 2mr \sin(\dfrac{d\theta}{2}) \cos(\theta + \dfrac{d\theta}{2}) \\ dy = 2mr \sin(\dfrac{d\theta}{2}) \sin(\theta + \dfrac{d\theta}{2}) \end{cases}$$

Since the model used in this paper is arc, it requests that the route passed between two near samples is arc-like circuit. Thus a high enough sampling frequency is needed by the system. In order to meet this kind of necessity, high-speed microprocessor PC104 was introduced into the system. The robot works fairly well for the excellent performance of PC104.

Moreover, if the arc model is true for high-quality sampling signals, the result of $d\theta$ will be definitely reliable, because there are not approximate computations in the formula. Oppositely the answers of dx and dy are not always available for the similar processing, when calculating sine and cosine, so it is important to constructing the engine for revision according other environments.

References

[1] Kortenkamp D, Bonasso RP, Murphy R (1998), Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot System, MIT Press, March 1998.

[2] Cassandra A, Kaelbling LP, Kurien J (1996), Acting under uncertainty: Discrete Bayesian Models for Mobile-robot Navigation, *Proceedings of IEEE RSI Int. Conf. on Intelligent Robot and Systems (IROS-96)*, 1996.

[3] Taylor PM (1990), Robotic Control, Macmillan Distribution Ltd., 1990.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

The running experiment of the wheel type mobile robot

Masanori Sugisaka and Hisashi Aito
Department of Electrical and Electronic Engineering Oita University 870-1192 Japan
Tel 001-81-97-554-7831, Fax 001-81-97-554-7841
E-mail msugi@cc.oita-u.ac.jp
haito@cc.oita-u.ac.jp

Abstract

In this paper, it used a soccer robot which needs the important field of robot technology as the wheel type mobile robot. With the soccer robot, as for the especially important one, "strategy" "the orbit control of the robot", and "the efficiency of the robot" is given. Therefore, it paid attention to " the orbit control of the robot " and it controlled an orbit of the soccer robot using the PID control.

1. Introduction

Recently, a lot of engineers aims to make a robot with human society and the more flexible relationship which can be applied in the wide area. Robotics/Intelligent machines has become a matured and accepted area of technology in developed countries. The research on robotics is related to the needs of the scientific community, industrial and governmental policies/strategies, social needs, availability of labor and the international competitiveness. The evolutionary advancements in machines started with the industrial revolution in nineteenth and twentieth centuries as a means of exploring variety of workplace aids. The efficiency and reliability of machines which could surpass the human physical capabilities. With the advent of theories on Artificial Intelligence a totally new era of intelligent/thinking machines surfaced. In this study, we report controlling an orbit by the soccer robot which consists of multi -agent system.

2. Soccer robot system

As for the method of the control of the soccer robot system, the one based on the sight is general. Basically, robots, a vision system, a host computer and a communication system are needed for robot soccer game as in Figure 1. It takes a field with the camera. Then, it is the one to process the picture in computer and to send an order to the robot with the radio. Three operating methods could be considered for a soccer game using this system: remote-brainless soccer robot system, vision-based soccer robot system and robot-based soccer robot system.

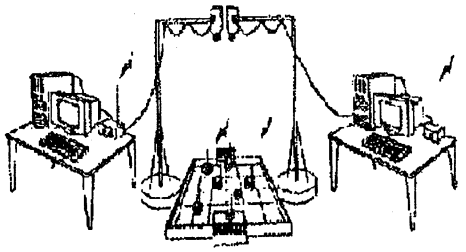


Fig.1 Soccer robot system

2.1 Distinguishing system

There are various ways in the distinguishing recognition system which detects object one. In this study, we fix the color of the object one which becomes a goal using RGB values. Then, it computes the position and the direction of the ball and the robot by the information.

2.2 Structure of the control

Figure 2 shows a basic control structure of the soccer robots. Robot hardware architecture is described in the dotted line box, and the robot software is represented in a solid line box. Each robot has the same control structure except the specialized intelligence part. The specialized intelligence part consists of each robot's own special behavior or strategy. The basic behaviors are "move",

"obstacle avoidance" and the basic actions are "shoot", "position-to-shoot", "intercept ball", "sweep ball" and "block". Theses are important behaviors and actions in a robot soccer game.

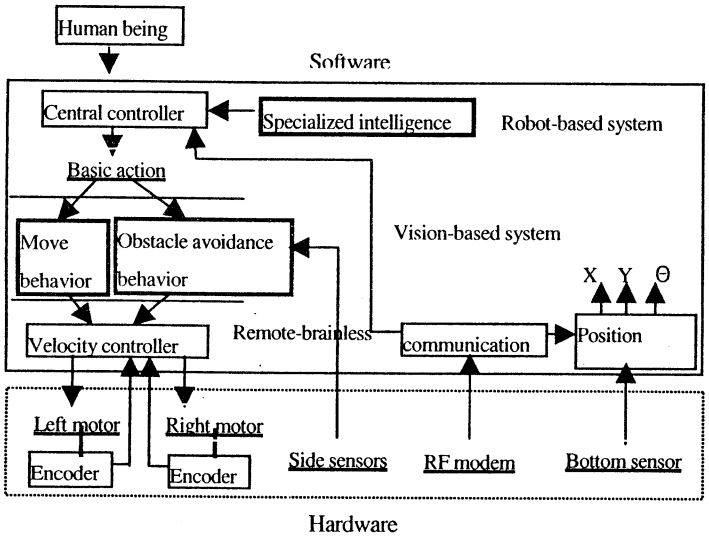


Fig.2 Control structure of the soccer robot

2.3 Basic behavior of the robot

The robot stands up basically with the operation of two movements, of turns, forward and back. It approaches a goal point while repeating turns, forward and back if order communication is transmitted to each robot from operator and host computer. The way of making a robot head for the goal point computes the inner product of the vector to the present robot direction of the progress and the vector to the direction of the goal and repeats it. Also, by which boundary the goal point is in the robot which reacts to it is different. Then, it advances towards the goal one while repeating revision from the big error to the small error and the error is a maximum of 1 degree. If supposing that the goal point is behind the right of the robot, the robot turns optionally in the left. When rotation mode is complete, the robot advances for the goal while it repeats the revision of the error and changes the speed of either side wheel. The same thing is desirable for the speed of the wheel on either side. In case of being different speed, it doesn't become a smooth movement. The robot stops when not being in the condition which chases a ball like loses sight of the ball.

2.4 Shoot action

All robots do the action which is various respectively , most are usually attack action. When giving the position information of the ball and the goal, two pieces of position relation are computed. "shoot action" is shown in figure 3 and "position-to-shoot" is shown in figure 4. "shoot action" needs two conditions which the ball is between the robot and the goal, and it sees a ball from the robot and there is a goal on the straight line (in this case Θe<Θb<Θh). The move of defender robot which tries to prevent a shoot is shown in case2 of figure 5.

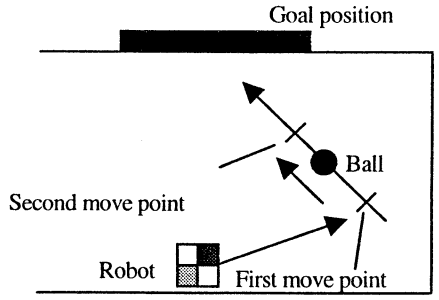


Fig.3 Shoo and position-to-shoot actions

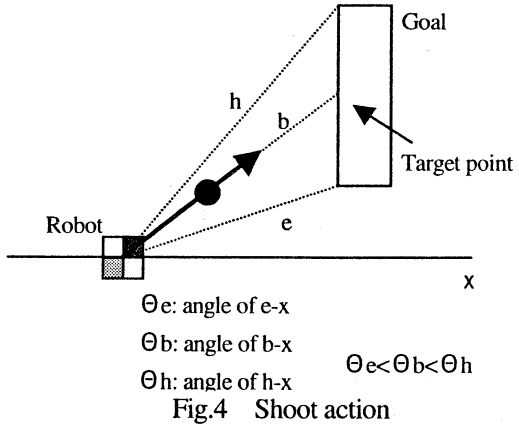


Fig.4 Shoot action

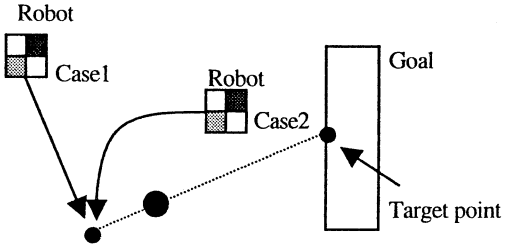


Fig.5 Position-to-shoot action

3 Orbit control of the soccer robot

3.1 Orbit formation of the robot

The way of making a robot head for the goal point is to repeat it which computing the inner product of the vector to the present robot direction of the progress and the vector to the direction of the goal point. When there is an error of the goal point, it advances while repeating the revision of the error.

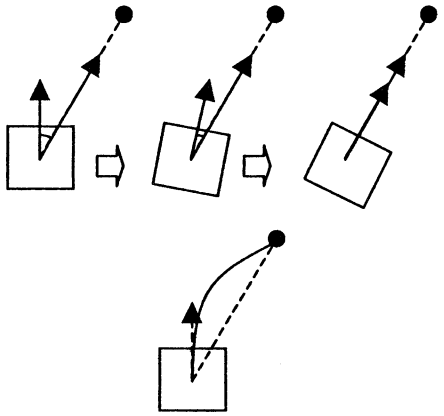


Fig.6 Orbit formation of the robot, ideal orbit

3.2 Orbit estimate of the ball

By the orbit of ball's moving like a straight line as shown in figure 7, it makes to be $y = ax + b$.

Then, the coordinate which will reach the Δt second of the ball from the following formula can be estimated.

$$Ax^2 + 2Bx + C = 0$$

$$A = (R-1)(a^2+1)$$

$$B = R\{-X_r + a(b-Y_r)\} + X_b - a(b-Y_b)$$

$$C = R\{X_r{}^2 + (b-Y_r)^2\} - X_b{}^2 - (b-Y_b)^2$$

$$R = V_b{}^2 / V_r{}^2$$

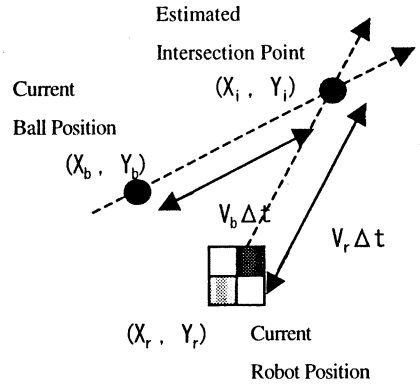


Fig.7 Orbit estimate of the ball

3.3 PID control

It is using PID control for the speed control of the robot, and the basic form is shown in figure 8. There is one number of the control quantity, the operation quantity, outside disorder. That is, it limits to the scalar control system. It supposes that outside disorder is added to the control object in the form which is added to the operation quantity.

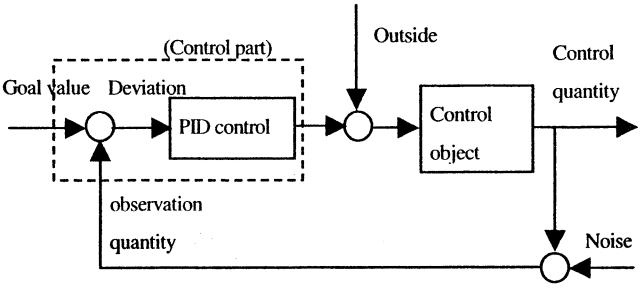


Fig.8 Basic form of the PID control

The control law of the PID control becomes

$$u(n) = K_p * e(n) + K_i \sum_{n=0}^{n} e(n) + K_d [e(n') - e(n'-1)]$$

when it shows a deviation in $e(t)$.

K_p is called proportional gain, K_i is called the integral time and K_d is called the differential time.

Proportional gain is a ratio with deviation and operation quantity in case of proportional operation. In the integration time, it is equal to the time that the output of the proportional operation and the differential calculus

operation to the same figure of lamp input becomes equal.

4 Experiment

In this experiment, it made go straight on to the goal point which left a robot by 100 cm to the direction of the moving of the robot like figure 9.

Then, it made advance towards the goal point of 45 degrees from the direction of the moving of the robot while changing the parameter numerical value of P, I, D. The robot corrects several times in the direction by the time it arrives at the goal point. It presented the evaluation of the experiment in the number of times of the directional correction and it measures the error of goal point and the spot which the robot stopped with the eye.

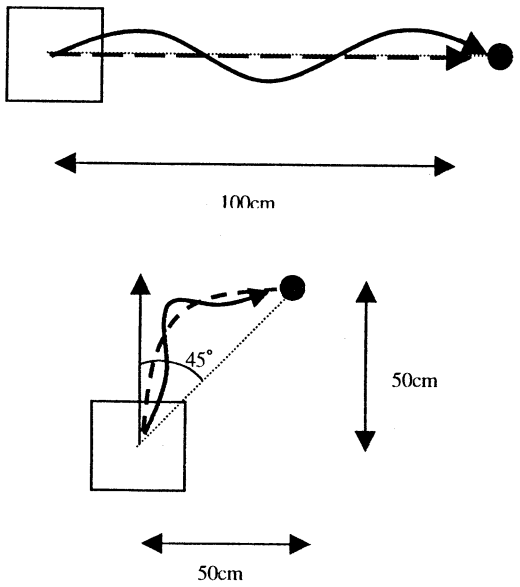


Fig.9 Ideal orbit and actual orbit

5 Simulation result and Conclusion

In this time, first, it found the parameter value of the best evaluation by the P control. Second, it gave from 0.0 to 30.0 a value to each parameter value the PI control, the PID control and it did it 10 times at each value. As a result, when a robot was normally worked, it did orbit correction with 1.2 times and an error with the goal point was settled within about 10 cm. Each parameter value when getting to be nearest the ideal orbit after repeating measurement became K_p =6.8, K_i =2.7, K_d =3.9 respectively. Also, the time when the miss- operation such as advancing towards another direction completely happens fitted the goal point and suddenly begins a turn in the place. Therefore, it sometimes was being difficult to let out an experiment result, too.

This experiment made a parameter value from 0.0 to 30.0. However, it thinks that it gets to be nearer the ideal orbit than with the result this time when taking numerical value, being smaller.

Also, the operation time to the goal point and so on for the difference with the excellent robot has a big difference. It thinks that the difference is the way of the operation control. Therefore, it increases the number of times of the operation to the unit time how, it is possible to do process time to the goal point shortly how much and it is a problem in the future.

6. References

[1] M.Sugisaka, "The system theory and the control"in Japanese, Nikkankougyou Shinbunsya, 1997

[2] S.Kitamura, O.Katai, "The picture and the control" in Japanese, Asakura Syoten. Co.,Ltd ,pp.160-pp.170,1991

[3] C. R. Kuba and H. Zhang, "The Use of Pereceptual Cues in Multi-robot Box-Pushing," IEEE Proc. Int. Conf, Robotics and Automation, pp. 2085-2401, 1996

[4] N. Mitsumoto, T. Hattori, T. Idogaki, T. Fukuda and F. Arai, "Self-organizing Micro Robotic System (Biologically Inspired Immune Network Architecture) and Micro Autonomous Robotic System" IEEE Proc. Int. Sym. On Micro Machine and Human science-Toward Micro-Mechatronics, pp. 261-270, Japan, 1995

[5] H.-s. Shim, M.-J. Jung, H.-S. Kim, I.-W. Choi, W.-S. Han and J.-H. Kim, "Development of Vision-Based Soccer Robots for Multi-agent Cooperative Systems", Proceedings MIROSOT' 96, pp. 29-35, Nov. 1997

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001.

Reinforcement Leaning Using a Gauss-Sigmoid Neural Network

Shin'ichi Maehara, Masanori Sugisaka, and Katsunari Shibata
Department of Electrical and Electronic Engineering, Oita University
700Dannoharu, Oita 870-1192, Japan. Email: maehara@cc.oita-u.ac.jp

Abstract

Boyan et al. has pointed out that the combination of reinforcement learning and Sigmoid-based neural network sometimes leads to instability of the learning. In this paper, it is proposed that a Gauss-Sigmoid neural network, in which continuous input signals are put into a Sigmoid-based neural network through a RBF network, is utilized for reinforcement learning. It is confirmed using simulation of the same task as in Boyan et al.[1] that the learning is faster and more stable when the Gauss-Sigmoid neural network is used, than when the Sigmoid-based neural network is used.

1 Introduction

Recently, the autonomous ability of reinforcement learning has attracted public's attention for the development of autonomous robots and learning machines. Reinforcement learning is generally used for action planning, and the machine learns the mapping from each state in the designed state space to an appropriate action.

By the combination of reinforcement learning and neural networks, a series of processes from sensors to motors including recognitions, can be synthetically learned[2]. Also, it is possible to acquire a continuous state space through learning in the hidden layers of the neural network, which can be utilized in another task.

However, Boyan et al. has pointed out that the combination of reinforcement learning and Sigmoid-based neural network sometimes leads to instability of the learning[1]. On the other hand, Gordon and Sutton showed that the instability of learning could be avoided by employing approximation methods based on the localization of input signals, such as CMAC, k-nearest-neighbor, and RBF (Radial Basis Function) [3][4][5]. If a task needs approximation of a strong non-linear function, then localizing continuous signals and using a representation like a table-look-up is effective. However, the output of such functions is represented as a linear sum of the localized signal, and they don't have hidden layers to represent global information, by integrating the localized signals adaptively. For example, when robots learn more than one task, the knowledge, which could be obtained from the previous sets of learning, is not utilized in present learning. Gaussian soft-max network[6] also utilizes RBF (Gaussian) units, and its generalization ability is an improvement on the regular RBF network. However, performance is not improved in areas where the RBF units are densely assigned.

In this paper, we use the neural network called the Gauss-Sigmoid neural network[7] and verify the stability in reinforcement learning for the hill-car task that Boyan et al. employed[1].

2 Hill-car problem

In this section, the hill-car problem is introduced as a problem in which the approximation of strong non-linear function is required. In this task, a car is located somewhere on the slope as shown in Fig.1, and must go up the right slope.

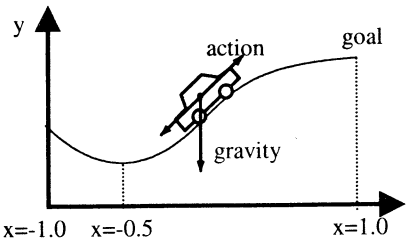


Fig.1 hill car problem.

The equation of the slope is described as

$$\begin{cases} y = x(x+1) & \text{if } x < 0 \\ y = x/\sqrt{1+5x^2} & \text{if } x \geq 0. \end{cases} \quad (1)$$

The equation of the car's motion can be described as

$$\begin{cases} \dfrac{dv}{dt} = \{\dfrac{action}{m} - \dfrac{y'}{\sqrt{1+y'^2}} g\}/\sqrt{1+y'^2}, \\ \dfrac{dx}{dt} = v, \quad \dfrac{dy}{dx} = y', \end{cases} \quad (2)$$

where x, y: position of the car, m: mass of the car, *action*: driving force of the car g: gravity. In this problem, since the maximum driving force of the car is not so strong, the acceleration in the climbing direction around the steepest area of the slope is always negative, Therefore, when the car starts from (x, v)=(-0.5,0.0), the car cannot climb the right slope at one go even if it tries to climb with the maximum driving force. When the car is located on the right side of the boundary whether the car can climb the right slope at one go or not, the direction of the driving force should be right. When it is located on the left side of the boundary, the direction of driving force should be left because the car has to go up the left slope and then returns to the right slope with higher velocity. Thus, at this boundary, both the ideal value function and action function become discontinuous, and the approximation of these functions needs strong non-linearity.

3 Gauss-Sigmoid neural network

Sigmoid-based neural network (NN) have global generalization ability, since the output function of each unit is sigmoid function, but it is not suited for approximation of strong non-linear functions such as a step function. The output of RBF network is represented as a linear sum of the localized signal, however they don't have hidden layers to represent the global information by integrating the localized signals adaptively. Therefore, in this paper, we employ the Gauss-Sigmoid NN in which the output of RBF is used as the input of the hidden layer of sigmoid NN as shown in Fig.2.

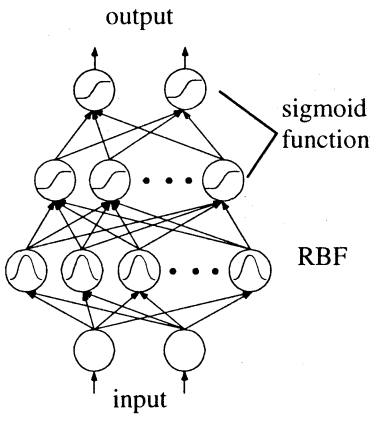


Fig.2 Gauss-sigmoid Neural Network.

Sigmoid NN itself is weak in the approximation of strong non-linear functions, but it becomes easy to approximate a strong non-linearity by using localized signals as inputs. Moreover, it is possible to obtain the global representation, through learning, by integrating the localized signals.

The output of RBF network is written as

$$output = \sum_{d=1}^{D} w_i g_i(x) + \theta, \quad \text{where} \tag{3}$$

$$g_i(x) = \exp(-\frac{1}{2}\sum_{d}^{D}(\frac{x_d - \mu}{\sigma})^2) \tag{4}$$

where $(\mu_{i,1},...\mu_{i,D})$: the center of the i-th RBF unit, $(\sigma_{i,1},...\sigma_{i,D})$: the size of the i-th RBF unit, θ : bias, n: the number of RBF units, D : the number of the input patterns. Both parameters of each RBF unit are also trained by the back-propagation learning method (BP), in the same way as the weights. However, when the size σ, is too small, the shape becomes steep and the updated value becomes too large. Then σ is converted to a logarithmic scale, and the update value is multiplied by the size σ. The update equation of the center μ is written as

$$\Delta\mu_{j,d} = g_j(x)\frac{x_d - \mu_{j,d}}{\sigma_{j,d}}\delta_j, \tag{5}$$

where δ_j : propagation error. In the training of size σ, s is defined as

$$\sigma_{j,d} = \exp(s_{j,d}), \tag{6}$$

and is updated as

$$\Delta s_{j,d} = g_j(x)\frac{(x_d - \mu_{j,d})^2}{\sigma_{j,d}^2}\delta_j. \tag{7}$$

Then s is transformed into the size σ by Eq.(6).

4. Actor-critic architecture

Q-learning and Actor-critic architecture are known as popular reinforcement learning algorithms. Here, Actor-critic architecture is employed as the paper of Boyan et al.. Actor-critic architecture is composed of an actor (action generation part) and a critic (state evaluation part). The critic evaluates the present state based on the past experiences of the system, and actor learns the action signal. In the critic, the previous state value is updated by using the present value to decrease TD (Temporal difference) error

$$\hat{r} = r_t + \gamma\ P(x_t) - P(x_{t-1}), \tag{8}$$

where γ : a discount factor, r: reward, x_t : input, $P(x_t)$: value. The update equation of the state value is written as

$$\Delta P(x_{t-1}) = \alpha_p \hat{r}_t, \tag{9}$$

where α_p : a learning rate of the value.

On the other hand, in the actor, $a(t)$ is the output and the actual action signal $\tilde{a}(t)$ is chosen from a stochastic distribution whose center is $a(t)$. Then, the action is updated to obtain more gain of the state value.

$$\Delta a(x_{t-1}) = \alpha_a(\tilde{a}_{t-1} - a(x_{t-1}))\hat{r}_t, \tag{10}$$

where α_a : a learning rate of the action. In this

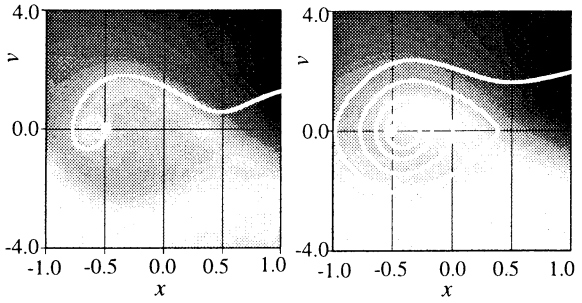
simulation, only one Gauss-Sigmoid NN is used for both the actor and critic. The network has two output units, one is for the action, and the other is for the value. If the action is not a scalar, the number of action outputs equals the number of elements of the action vector.

5. Simulation

The hill-car problem is solved by actor-critic type reinforcement learning. Learning ability is compared in three cases; (1) Sigmoid-based NN, (2) Gauss-Sigmoid NN, and (3) RBF network. The initial state of the car is chosen from random numbers within the limits of $-1<x<1$ and $-4<v<4$. The actual driving force is the sum of the action, and the small uniform random number powered by 3. The state transition is calculated by solving the Eq. (2) by the Runge-Kutta method. When a car arrives at the top of the hill, the reward $r=1.0$, otherwise $r=0.0$. In the critic, the state value is updated by Eq. (9), but when the car rushes out to the left side of the slope or arrives at the top of the hill, $p(x_t)$ in Eq. (8) is 0. The velocity of the car was fixed at -4.0 when it became smaller than -4.0, and 4.0 when it became larger than 4.0. Since the driving force, *action*, was limited from -3.0 to 3.0, the car must go through the state of $x \leq -0.74$ with $v=0.0$ on the left side to arrive at the top. The value range of the sigmoid function is from -0.5 to 0.5. The Gauss-Sigmoid NN and Sigmoid-based NN used both a single hidden layer, and 40 neurons. The number of Gaussian units in the RBF network and Gauss-Sigmoid NN is 110(11×10). The learning of center and size of the Gaussian units aren't performed in this simulation. The learning rate for each network is 100/sqrt (the input number of each unit) in Gauss-Sigmoid NN, 80/sqrt (the input number of each unit) in Sigmoid-based NN, 0.48 in RBF network, and a momentum term was not used. The learning was iterated for 300,000 steps.

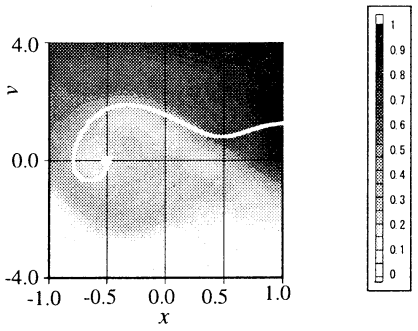
The value functions and the loci of the car in the hill-car task are shown as Fig.3. The initial state of the locus is (x, v)=(-0.5,0.0) which is the bottom of the slope. In all cases, the value is large in the upper right part where x and v are large. The value surface has a cliff around the boundary, whether a car can climb right slope at one go or not. The ridge of value function can be observed clearly around (x, v)=(-0.85,0.0), and the car can climb top of the hill after it goes up the left slope in the case of Gauss-Sigmoid NN and RBF network. While in the case of Sigmoid-based NN, the ridge of value function cannot be observed clearly, and the car cannot climb the slope unless it goes back and forth many times

Fig.4 shows that driving force as a function of the car's states. In the case of Gauss-Sigmoid NN, the boundary where the direction of the driving force changes can be clearly. It can be known that the direction of the driving force is left when the car is the lower part of the left slope with a small negative



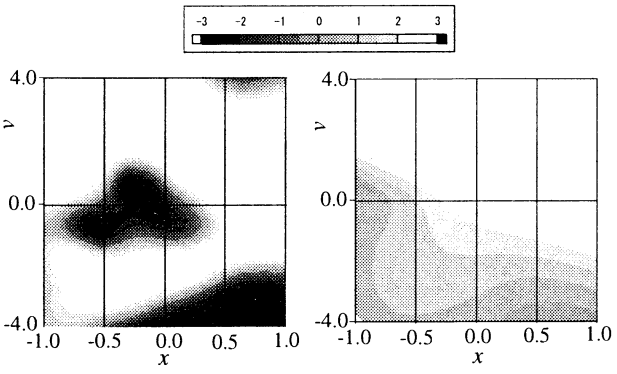
a) Gauss-Sigmoid NN (11×10 Gaussian) (40 sigmoid units)

b) Sigmoid-based NN (40 neurons)



c) RBF network (11×10 Gaussian)

Fig.3 Value function and locus of the car in hill-car task. The initial state of the locus is (x, v)=(-0.5, 0.0) that is the bottom of the slope.



a) Gauss-Sigmoid NN (11×10 Gaussian) (40 sigmoid units)

b) Sigmoid-based NN (40 neurons)

Fig.4 The magnitude and direction of the driving force.

velocity and when the car is on the lower part of the right slope with a small velocity. While, in the case of Sigmoid-based NN, there are no clear regions where the direction of the driving force is left.

The learning curve in hill-car task is shown in Fig.5. The average number of steps to the goal over 37 initial states is plotted every 2000 steps. The 37 initial states are located on the grid at intervals of (Δx, Δv)=(0.25,

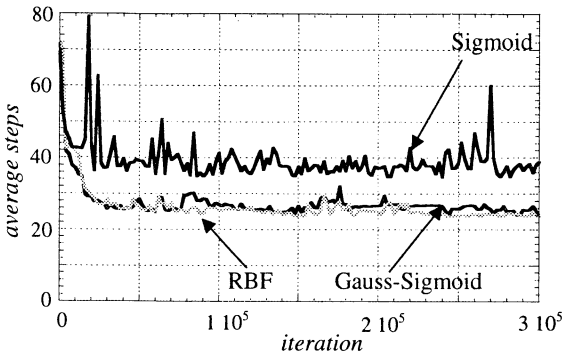


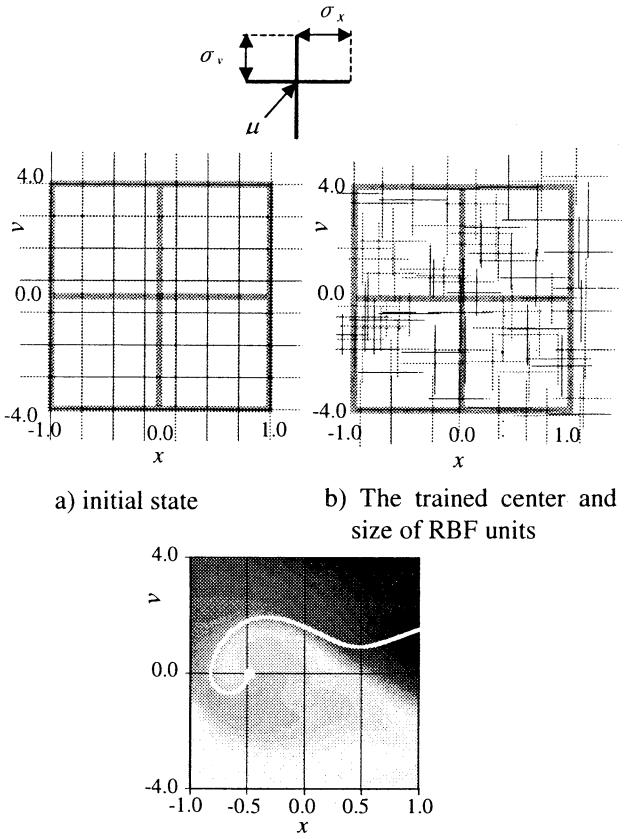
Fig.5 Comparison of the learning curve in the hill-car task.

1.33). The other 11 states, from which the car cannot climb the slope physically, are excluded. In Fig.4 it is shown that the learning speed and error of Gauss-Sigmoid NN is almost equal to RBF network, even though the sigmoid function is utilized. The error did not decrease as well in the Sigmoid NN. In the Sigmoid NN, the error did not change when the number of the hidden neurons was increased to 100.

Next, the center μ and size σ of each RBF unit is trained in the Gauss–Sigmoid NN. By this process, it is expected that efficient learning is performed and the same learning performance can be realized with fewer RBF units. Fig.6 shows the center and size of each of RBF unit, after learning, in the case of the Gauss-Sigmoid NN. The number of RBF units in this Gauss-Sigmoid NN is 64(8×8). The range of input signals is shown by a gray-framed rectangle in each figure. It is clear that many RBF units move to the place where strong non-linearity is required, and their sizes become small. When the parameters of RBF units were not trained, it could not reach the top at one go in all the 3 simulations in which the initial weights are varied, while it could reach in all the 3 simulations when the parameters were trained.

6. Conclusion

The paper proposed use of the Gauss-Sigmoid neural network, when the input signals represent continuous and global spatial information. In the hill-car task, it was shown that the performance was almost the same as RBF network and better than the as that of an performance of a Sigmoid-based neural network. Furthermore it was also shown that by learning the parameters of RBF units, the task could be solved with fewer RBF units.



a) initial state

b) The trained center and size of RBF units

c) Value function and locus of the car in case of moved RBF unit (8×8 Gaussian)

Fig.6 The change of center and size of RBF units, and Value function and locus of the car when the parameters of each RBF unit was trained in the Gauss-Sigmoid NN.

Reference

[1] Boyan, J.A. & Moore, A.W. : Generalization in Reinforcement Learning : Safely Approximating the Value Function, *Advances in Neural Information Processing Systems*, MIT Press, 7, pp.369-376(1995)

[2] Shibata, K. , Ito, K. & Okabe, Y. : Direct-Vision-Based Reinforcement Learning in "Going to an Target" Task with an Obstacle and with a Variety of Target Sizes, *Proc. of Inter. Conf. on Neural Networks and Their Applications '98*, PP.95-102(1998)

[3] Gordon, G. J. : Stable Function Approximation in Dynamic Programming, *Proc. of the 12-th ICML*, pp.261-268 (1995)

[4] Sutton, R. S. : Generalization in Reinforcement Learning: Successful Examples Using Space Coarse Coding, *In Advanced in Neural Information Processing System*, vol8, pp.1038-1044 (1996)

[5] Sutton, R.S. and Barto, A.G. : Reinforcement Learning, The MIT Press(1998)

[6] Morimoto, J. , Doya, K. : Learning Dynamic Motor Sequence in High-Dimensional State Space by Reinforcement Learning –Learning to Stand up-, *Proc. IEICE*, J82-D- II ,No.11, pp. 2118-2131(1999) (in Japanese)

[7] Shibata, K. and Ito, K. : Gauss-Sigmoid Neural network, *Proc. of IJCNN'99*, #747(1999)

Author Index

[A]

Abe, K. 109, 208, 478, 528
Adachi, S. 387
Adachi, T. 309
Agarwal, S. 188
Aibe, N. 89
Aita, T. 365
Aito, H. 558
Aoi, S. 421
Aoyama, T. 486
Arimizu, T. 31
Arita, T. 37
Asakura, T. 456
Asharif, M. R. 305
Azuma, N. 58

[B]

Bae, H. 5
Bae, J. I. 272
Bao, G. 224
Barrett, C. L. 34
Berry, R. A. 536, offprint
Bickhart, M. offprint
Bode, M. 266
Breithaupt, R. 266
Bubnicki, Z. 220
Buller, A. 146, 150

[C]

Cao, G. 496, 500
Cha, S. M. 272
Chen, H. 504
Chen, Y. 204
Chodakowski, T. 146
Choi, J. Y. 272
Chung, M. J. 196

[D]

Dahlstadt, P. offprint
Danjoh, T. 113

[E]

Eggenberger, P. 152

[F]

Fatehi, A. 528
Fischer, J. 266
Franklin, S. 280
Fujita, H. 317
Fukuda, T. 142
Funamori, M. 62

[G]

Garcia, Y. 41
Girija, P.N. 482
Goto, S. 474
Gunji=Pegio, Y. offprint

[H]

Habib, M. K. 232
Hajiri, K. offprint
Hama, K. 539
Han, S.H. 13
Haneda, H. 337
Hashimoto, H. 13
Hasida, K. offprint
Hatono, I. 160, 180
Haw, C. offprint
Hernandez, G. 41
Hirasawa, K. 504, 516, 520, 470
Hirayama,H. 427, 431
Honda, H. 443
Honma, N. 478
Horiguchi, Y. 415
Horiuchi, T. 405
HoseinNezhad, R. 305
Hosoe, S. 176
Hosoe, Shig. 458
Hu, D. 329
Hu, J. 504, 516, 470
Husimi, Y. 365

[I]

Ichiguchi, N. 409
Ichiki, T. 62
Ida, M. 258
Igarashi, H. 296
Ikeda, T. 54
Ikegami, T. offprint
Imai, M. 50
Imai, K. 377
Inamoto, T. 172
Inoue, A. offprint

Inoue, K. 337
Inoue, M. 19
Ishigaki, S. 258
Ishiguro, H. 50
Ishii, H. 409
Isokawa, T. 387
Isurugi, Y. 447
Ito, Ka. 345
Ito, Ko. 101, 200
Ito, Ma. 284
Itou, Mi. 85
Ito, N. 353
Izumi, K. 126, 130, 134, 138, 142

[J]
Jeong, D. Y. 13
Jeong, M. S. 254
Jia, S. 105
Jiang, R. 329
Jin, T. S. 228
Jinguuji, T. 70

[K]
Kaiser, L. 146
Kakazu, Y. 184, 543
Kamaya, H. 208
Kang, Y. 93
Kashima, T. 447
Katai, O. 258, 401, 405
Katoh, R. 113, 121, 443
Kawaguchi, T. 547
Kawai, N. 73
Kawaji, S. 204
Kawakami, H. 401, 405
Kawakami, M. 421
Kawazoe, Y. 9
Khouri, K. 41
Kiguchi, K. 126, 130, 134, 138, 142
Kikuchi, K. 313
Kim, D. Y. 196
Kim, H. D. 13
Kim, N. 93
Kim, S. 5
Kim, S. I. 54
Kim, S. H. 272
Kitagaki, K. 466
Kitamura, S. 164, 172
Kitazoe, T. 54, 58, 62, 66
Kobayashi, K. 361
Kojima, K. 101
Komaki, D. 409
Komeiji, Y. 349
Kondadadi, R. 280
Kondo, E. 462
Kryssanov, V. V. 172
Kubo, M. 192, 216
Kudo, H. 543
Kusumoto, Y. 142
Kyura, N. 474

[L]
Lee, H. C. 272
Lee, H. Y. 208
Lee, J. J. 77, 97
Lee, J. M. 228, 254
Lee, M. H. 5, 13, 272
Li, S. 224
Li, Z. 224, 555
Liu, J. Q. 397
Loukianov, A. 156, 238
Luo, Y. 329
Luo, Z. W. 458

[M]
Maeda, M. 381
Maehara, S. 562
Maekawa, T. 73
Maeshiro, T. 27
Matsui, N. 387
Matsumura, T. 341
Matsuno, F. 345
Matsuno, K. 532
Matsuura, N. 81
McCaskill, J. S. 393
Mikami, S. 539
Minami, M. 456
Mitsui, T. 276, 292
Mitsuo, N. 551
Mizoguchi, F. 117
Mizuhara, H. 23
Moldovan, D. 524
Morikawa, K. 188
Morita, H. offprint
Morita, Ke. 377
Morita,Kaz. 321
Mortveit, H. 34
Moshiri, B. 305
Murao, H. 164
Murata, J. 504, 516, 520, 470

[N]
Nagai, Y. 250
Naitoh, K. 357
Nakamura, A. 466
Nakamura, Ma. 341
Nakamura, M. 474
Nakanishi, D. 180
Nakanishi, K. 551
Nakatsu, R. 45
Namatame, A. 192, 216
Nanayakkara, T. 126
Ninagawa, K. 212
Nino, F. 41
Nishimura, H. 58
Nishimura, J. 381
Nishina, E. 73
Nishiyama, H. 117
Noort, D. V. 393
Nowak, A. 146

[O]
Obayashi, M. 117
Odanaka, T. 31
Odashima, T. 458
Ogasawara, T. 466
Oh, S. K. 77
Ohi, N. 27
Ohkura, K. 168
Ohnishi, K. 490
Ohnishi, N. 543
Ohtsuka, Y. 543
Ohuchi, A. 81, 532
Oka, N. 188
Okada, M. offprint
Okada, N. 462
Okada, T. 333
Okita, Y. 427, 431
Okuhara, K. 19, 317, 333
Onaga, K. 341
Onat, A. 421
Ono, T. 50
Onodera, H. 250
Oohashi, T. 73
Oono, Y. 369
Oya, M. 443

[P]
Park, H. K. 196
Park, J. W. 254
Pei, J. 555
Peper, F. 387

[R]
Rao, P. S. 482
Reidys, C. M. 34
Rizon, M. 435, 547
Ryoo, J. R. 196

[S]
Sagara, S. 113, 121
Sagawa, Y. 212, 262
Saito, T. 23
Sakai, M. 478
Sakamoto, S. offprint
Sano, M. 109
Satoh, H. 192, 216
Sawaragi, T. 415
Sekiya, Y. 486
Seo, K. H. 77, 97
Serikawa, S. 321
Shiba, T. 81, 532
Shibata, J. 19
Shibata, K. 200, 562
Shibata, T. 276, 292
Shima, M. 447
Shimada, To. 242, 313
Shimada, T. 353
Shimizu, E. 284
Shimoda, H. 409
Shimohara, K. 27, 146 , 397
Shimomura, T. 321
Shin, Y. 93
Shinchi, T. 58
Shon, M. K. 520
Shutou, H. 490
Siwek, L. 1
Sogabe, T. 381
Song, J. S. 97
Suehiro, T. 466
Sugawara, K. 246,
Sugie, N. 212, 250, 262, 543
Sugihara, K. 66,70
Sugisaka, M. 85, 156, 200, 238, 288, 309, 551, 558 562
Sugita, Y. offprint
Sun, X. J. 496, 500
Suto, H. 401
Suzuki, K. 539
Suzuki, H. 373

Suzuki, N. offprint
Suzuki, Y. 325
Svinin, M. 168, 176
Szeto, K. Y. 329

[T]
Tabuse, M. 58, 66, 70
Tachibana, K. 301
Tagawa, K. 337
Taguchi, Y-h. 369
Takabayashi, J. 325
Takai, H. 301
Takase, K. 105
Tamaki, H. 160, 164, 172
Tamaki, S. 341
Tamamura, A. 456
Tamura, H. 486
Tamura, M. 113, 121
Tanaka, Hid. 284
Tanaka, Hir. 325
Tanaka, T. 19, 317, 333
Tanaka, Toshim. 250
Tang, Z. 486
Tanie, K. 276, 292
Toda, K. 401
Todaka, A. 66
Touda, A. 276
Tsuchiya, K. 421
Tsujita, K. 421
Tsukada, S. 242

[U]
Uchida, K. 250
Ueda, Ka. 160, 168, 176, 180
Ueda, K. 250
Ueno, O. 73
Umeo, H. 381
Umeyama, T. 212
Uno, K. 216
Uozumi, E. 262
Ushio, S. 176

[W]
Wada, K. 276, 292
Wada, M. 443
Wakamatsu, H. 439
Wan, W. 516
Wang, Q. 486
Wang, X. 224
Watanabe, K. 126, 130, 134, 138, 142
Watanabe, T. 109, 246
Wu, H. 224, 555
Wu, J. L. 23

[X]
Xiong, Q. 470
Xiong, S. 224

[Y]
Yamada, Kaz. 168
Yamada, Ko. 452
Yamada, T. 134
Yamagishi, H. 405
Yamamoto, A. 164
Yamamoto, H. 524
Yamamoto, M. 81, 532
Yamaoka, T. 37
Yamasaki, K. offprint
Yamashita, K. 321
Yan, L. 276, 292
Yanagawa, H. 361
Yasuda, G. 301
Yasunaga, M. 89
Yokoi, H. 184
Yoneyama, J. 512
Yoshida, A. 462
Yoshihara, I. 89, 109
Yoshikawa, H. 409
Yoshitomi, Y. 54
Yu, W. 184
Yukawa, S. 353

[Z]
Zainon, Z. 435
Zhang, T. 474
Zhang, X. 439
Zhang, Y. G. 288
Zhao, Q. 508
Zhou, Z. 224
Zhu, X. J. 496, 500

Proceedings of The Sixth International Symposium on

ARTIFICIAL LIFE AND ROBOTICS

(AROB 6th '01)

for Cognitive and Behavioral Intelligent Artificial Liferobot

OFFPRINT

January 15-17, 2001
U-Port, Gotanda, Tokyo, JAPAN

Editors: Masanori Sugisaka and Hiroshi Tanaka

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15–17, January, 2001,

The physical constraint programming for real emergent system

Koichiro Hajiri
IBM Tokyo Research Laboratory
1623-14 Shimotsuruma, Yamato-shi, Kanagawa-ken, 242-8502, Japan

Abstract

Here I show the only technical emergence program. To avoid ambiguity of natural language, I only put computer program that is infinite regression of Akires and Turtle paradox. In the program, there are no explicit description of halting, however, because of the limitation of expression of real number, the program stops. Thus, the state 'stop' has emerged from the infinite regression description.

1 The program

```
#include <stdio.h>
#include <math.h>

double akr(turtle, akires)
    double turtle;
    double akires;
{
    return turtle - akires;
}

double tur(ak_d)
    double ak_d;
{
    return ak_d/2.0;
}

main(argv, argc)
    int argv;
    char *argc[];
{
    double akires;
    double ak_d;

    double turtle;
    double tu_d;

    int time;


    /* init */
    akres = 0.0;

    turtle = 100.0;

    time = 0;

    while(akires < turtle){
ak_d = akr(turtle, akires);
tu_d = tur(ak_d);
turtle = turtle + tu_d;
akires = akires + ak_d;
time++;

printf("%d %le %le\n" , time, akires, turtle);
    }

}
```

2 The action

```
1 1.000000e+02 1.500000e+02
2 1.500000e+02 1.750000e+02
3 1.750000e+02 1.875000e+02
4 1.875000e+02 1.937500e+02
5 1.937500e+02 1.968750e+02
6 1.968750e+02 1.984375e+02
7 1.984375e+02 1.992188e+02
8 1.992188e+02 1.996094e+02
9 1.996094e+02 1.998047e+02
10 1.998047e+02 1.999023e+02
11 1.999023e+02 1.999512e+02
12 1.999512e+02 1.999756e+02
13 1.999756e+02 1.999878e+02
14 1.999878e+02 1.999939e+02
15 1.999939e+02 1.999969e+02
16 1.999969e+02 1.999985e+02
17 1.999985e+02 1.999992e+02
18 1.999992e+02 1.999996e+02
19 1.999996e+02 1.999998e+02
20 1.999998e+02 1.999999e+02
21 1.999999e+02 2.000000e+02
22 2.000000e+02 2.000000e+02
    .
    .
    .
52 2.000000e+02 2.000000e+02
```

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Mixing with Aliens:

Life and music on Gakki-mon Planet

Rodney Berry, Palle Dahlstadt, Catherine Haw
ATR Media Integration and Communications Laboratories
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, Japan.
rodney@mic.atr.co.jp
www.mic.atr.co.jp/~rodney

Keywords: artificial life, music, art, virtual worlds

Abstract: Gakki-mon Planet is a world where visitors can engage with musical creatures and plants as they explore an alien landscape. Inspired by processes in nature, the creatures evolve complex physical and musical behaviours. The visitor can use a joystick to move about the world and influence the creatures' breeding and feeding etc. A MIDI keyboard can be used to join in the creatures' music. It represents a new approach to musical composition, more akin to the personal expression to be found in gardening or keeping a pet.

The name Gakki-mon comes from *gakki* for musical instrument and *mon* for monster. Gakki-mon Planet is a kind of creative game that explores ways of making musical virtual environments on the computer. The main purpose of the project is to abstract processes found in nature, such as metabolism and heredity, and apply them to the composition of music. We have made a piece of music that you can move around inside to examine its various elements as if moving through an alien landscape. Although it is primarily intended as a work of art, it is hoped that virtual worlds populated by artificial life-forms may influence people to appreciate real living systems in a deeper way. Our work paves the way for a new kind of educational entertainment product providing an exciting form of creative expression. We see this expression as existing somewhere between the creative act of writing and playing music and the satisfaction of growing a garden or caring for a pet. As music surrounds the player in this virtual world, opportunities arise for interaction between the player and the creatures, through music and also between the creatures themselves. Gakki-mon Planet will eventually run on a home or school PC in real time and provide an enjoyable environment in which to learn about genetics, music, evolution, ecology etc.

Players or visitors to this planet explore the relationship between musical composition and artificial life. Players are introduced to a surreal landscape with food blocks, hills, valleys, lakes and percussion-sounding trees. The food blocks are randomly distributed about the planet A white, wire cube, as shown below in Figure 1.0, represents the player's

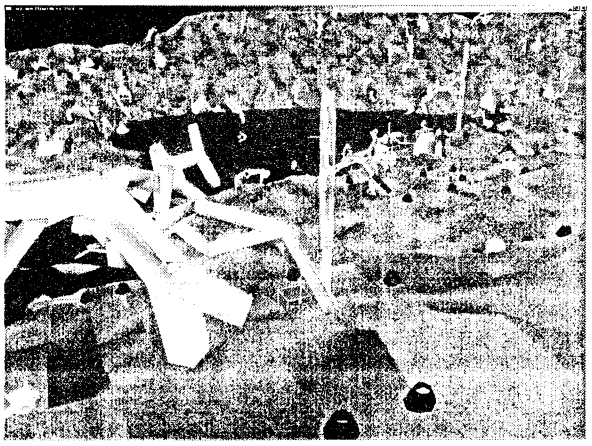


Figure 1 Gakki-mon Planet has trees, creatures, lakes and food blocks

position in the world. Using a joystick controller the player can maneuver the cube around the world to listen to and interact with the creatures. Apart from the joystick, the player can also use a MIDI keyboard.

The musical creatures in this world can walk, eat, mate, play music, die and evolve. In order to survive, creatures must maintain an energy level above zero; negative energy results in death. Eating will increase their energy while walking and mating will decrease their energy. Evolution occurs gradually over multiple generations through the mating process.

The genotypes of the creatures determine their appearance and behaviour both physical and musical. The creature's nervous system is made up of a genetically determined mathematically based neural network. This network is made up of blocks of mathematical functions each with inputs and outputs connected to other blocks. The various functions and their wiring are inherited from the creature's parents. The network processes incoming information form the creature's sensors then feeds it out to various effectors. Each frame, creatures are evaluated for each of their effectors. The first effector is used to determine the creature's direction that it will walk, and also to determine the MIDI note value for the creature. The remaining seven effectors are the muscles for the creature's trunk, as there are three trunk segments each

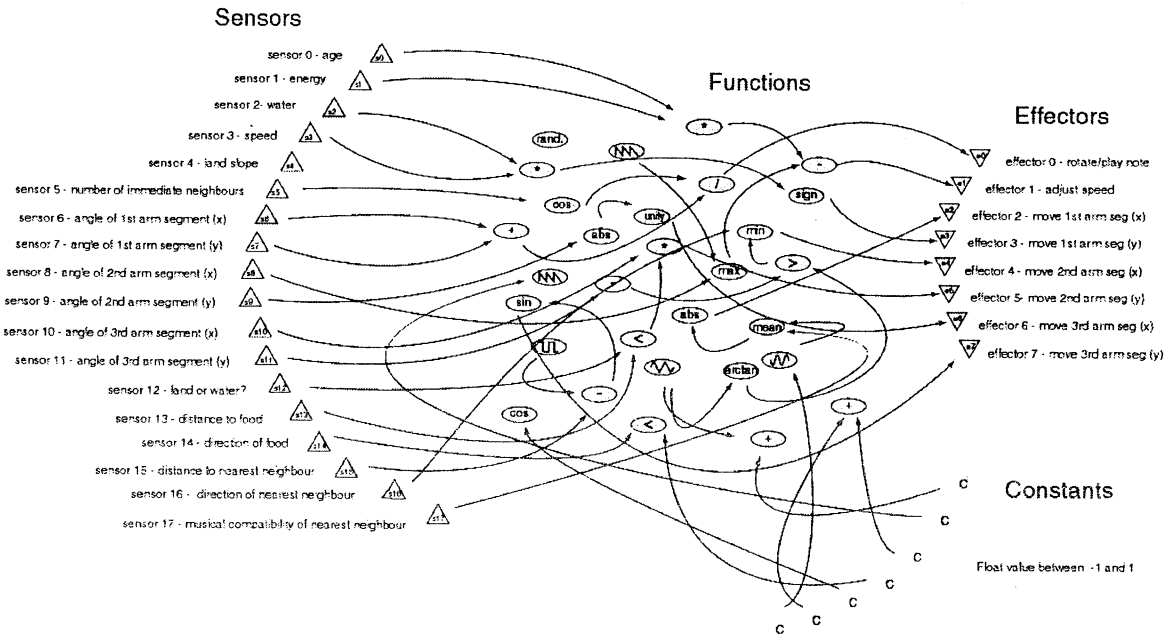


Figure 2 an example chromosome

requiring x and y values. One example of resulting complex behaviour might be for a creature not to seek food that is nearer if there are many creatures in the area but to instead seek food a little farther with fewer creatures, hence less competition. This sort of complexity is yet to be seen in Gakki-mon Planet. An example of a chromosomal wiring depicting the sensor values and effector details are shown above in Figure 2.

Musical elements are controlled by a separate array of 35 MIDI parameters that are also inherited from each of a creature's parents. This generates quite complex variations in musical output making it fairly easy to identify the sound of a particular individual. The volume level of each creature is controlled by the position of a virtual microphone associated with the white cube that indicates the user's position in the world. The sound can be made to come from any of four speakers creating the illusion of a spatialised sound environment.

Creatures also use music as a criterion for mating. Mating will occur if two creatures, which possess a minimum energy level, are within a specified distance and direction and are musically compatible. Creatures must eat the food blocks in order to increase their energy while walking and mating decreases their energy. Musical compatibility is defined by whether the creatures' have similar sounds to each other. In other words, the creatures' share similar MIDI Values. .

The offspring will be created as an exact duplicate of one of the parent creatures, referred to as the *mom* and then a crossover will occur with the *dad*. The crossover procedure is designed to increase the probability that distinctive traits will be inherited from the parents. For example, if the parent has the ability to find food then it's offspring can potentially inherit the same trait.

Mating is extremely important in Gakki-mon Planet, as it is the basis for evolution; there are no natural, spontaneous mutations or adaptations to the environment. In Gakki-mon Planet the creatures are evaluated based on their ability to find and eat food (i.e. the fitness function is to eat food). Therefore, with each generation the creatures will evolve to become better food seekers as only the "fit" creatures will possess the threshold energy level to mate and pass on their chromosomes to the next generation. However, like most genetic programs, evolution is a time consuming process that requires hundreds, even thousands, of generations in order to get significant results.

Trees in the world also have a primitive nervous system and are associated with percussive sounds in the music. The rhythm sequences of the trees serve to underpin the more erratic sounds of the actual creatures. At the time of writing, the music produced has been described as rather sparse bio-industrial music. Some people like it and some do not. Later I think it will be possible to produce a variety of styles of music. At present, the system favours music that uses lots of repetitive figures like modern electronica/techno dance music for example.

The player interacting with Gakki-mon Planet can use the joystick to select individual creatures. Once this is done, a creature can be moved to another

location, useful for giving them more food or sitting them down next to a pleasing group of trees or other creatures. Creatures can be induced to mate with another creature allowing the user to cultivate particular kinds of musical patterns. The player can also set the camera and microphone to follow a particular creature wherever it goes on Gakki-mon Planet. Following a creature allows a kind of musical narrative to develop. We hear the creature's musical sound as a constant pattern or ostinato that persists through the comings and goings of sounds from encounters with other creatures. A MIDI keyboard allows another level of interaction. At present, the sound controlled by the keyboard is linked to the genomes of the creature last selected. In this way the music played from the keyboard takes on different timbres as the player moves through the world. We intend to extend this interaction so that creatures can hear and respond to the music introduced by the player. In this way, the player becomes another creature in the virtual sonic environment.

We intend to extend Gakki-mon Planet in the following ways. Apart from the musical features described above, we are working to integrate the musical aspects in with the other aspects of the creatures' sensation and behaviour. This will lead to a closer relationship between what one sees and what one hears in the world. Music and movement naturally go together so, if the sensors and effectors of the creatures are directly connected to the music, they will engage in a kind of dance with the user and each other. We also intend to create more means to farm creatures by breeding them in separate enclosures or possibly by teleporting creatures to a separate world called *the* farm. It would also be possible to genetically engineer individual creatures in yet another virtual world called *the lab* by assembling their genomes by hand. Creatures resulting from the farm or the lab could then be released back into *the wild* to breed with the other creatures.

Finally, we believe that the combination of music and artificial life in a interactive virtual world provides an interesting form of personal expression. Within an artwork or game environment, a person can experience both creation and cultivation of living music that can challenge some conventional ideas of authorship. People playing with Gakki-mon Planet are not only making music themselves. They enter into a kind of caring symbiosis with the virtual life-forms in order to produce music that is hopefully quite different to that produced by directly deciding each musical event. It is possible that, by communicating with artificial life-forms in this manner, we may find that we are actually communicating with our selves and that the experience may somehow change the way we see our selves and our relationship to the real world.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

The problems under the dynamic environments need the consideration of the evaluation standard between the environments that are changing.

Kazuko Yamasaki
yamasaki@rsch.tuis.ac.jp
Tokyo University of Information Sciences.
1200-2 Yato town, Wakaba ward, Chiba city, Japan

Abstract- The method(The Dynamic Pareto Optimum GA) is proposed. This method makes it possible to take preference which concerns the dynamic environments into adaptation . It can support conservative behavior or short-term insight and so on.

1 Introduction

1.1 The Present Theory Lacks the Evaluation Standard through the Time.

We are always in the changing environments, and adapt against them to get more profit or to live more comfortable or to make ones dream come true. Among us, some people act conservative, and others act radical. And we usually select our action, under long-term insight or short-term insight, by case. Some people behave stably and others take high-risk instead of high-return. Someone is good at getting on the flow and someone likes to go against the flow. Their description are all concerning to how to behave in the changing environments. But recently developed evolutional computation lacks such a point of view. These algorithms always base contemporary fitness function. When fitness function changes, the agents in the environments can not compare this time situation with that time situation. It is impossible to make such algorithm support conservative behavior or short-term insight and so on. And if the environments are changed before convergence of population, the instability appears under usual selection method which has relative values among individuals. For example, the selection method with scaling , ranking selection. If the environments are changed cyclically at every generation, the population converges different point by each seed under ranking selection. Moreover sometimes it shows quite unstable behavior as I describe after. The reason of these results is The Present Theory Lacks the Evaluation Standard through the Time. Then I introduce pareto optimum which is used in the field of multi-objective optimization into the dynamic problem, make it possible to express conservative behavior or short-term insight and so on. Their consideration is necessary to adapt to the changing environments at the time scale slower than evolution. But it is not only this case. If the agents follow the changes, at the next change of environment, there is a tendency that sudden dive of fitness appears, in not only on-line performance but also off-line performance. But there are some problems for which this dive is not suitable. For example if the problem concerns money, the dive brings bankruptcy.

2 The Dynamic Pareto Optimum GA

Next to practice these ideas, I propose the Dynamic Pareto Optimum GA and some result of experiments. This method has the next three characteristics.
In order to bring the preference into method, I make the agents living through generations and keep fitness values in their memories. In my model, at every generation, 10% of agents in the population die and are given birth to, so agents can live through several generations.
An agent has chromosome and additional memory in which the values of fitness are kept as time series.
Regarding the different environments which appear one by one as the multi-objective optimization fitness functions, using time series of fitness in the agent's memory, we can determine dominant relationship among the agents.
Counting the number of agents who dominate an agent, we can rank all agent by multi-objective-ranking method.

2.1 algorithm

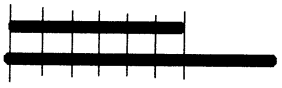
The algorithm consists of next items.

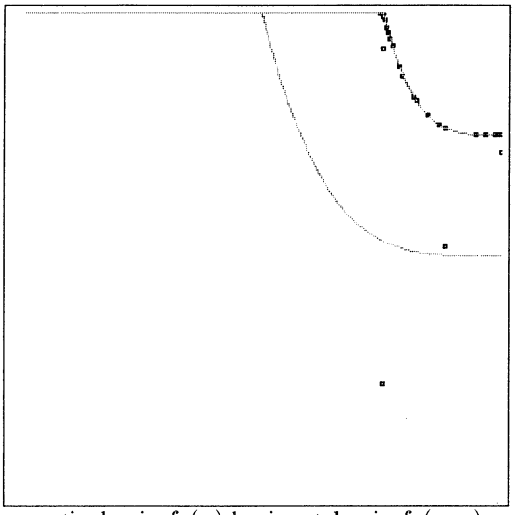
1. Multi-objective-ranking.
2. Share by distance in the adaptation space.
3. (Preference)
4. Select 10% of the population and reproduce and mutate
5. Return 1

The multi-objective-ranking consists of next items.

1. Select two agents who live more than 3 generations.
2. n is the generations in which the younger agent among the two agents has lived. Go back along the time series of two agents, from now to birth time. Compare the two time series for n generations. According to them, determine the dominant-relationship of two agents as follows.

   Agent A dominates agent B:
   all n generations fitness of A >= fitness of B
   more than a generation fitness of A > fitness of B





vertical axis $f_1(x)$ horizontal axis $f_2(x, y)$

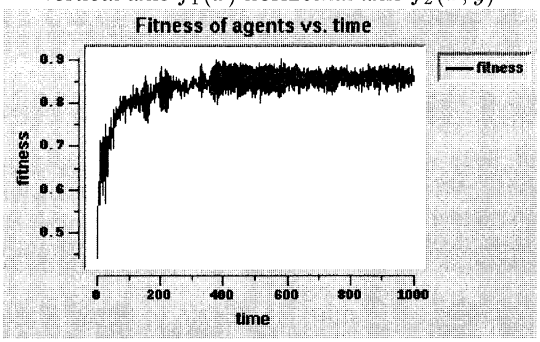
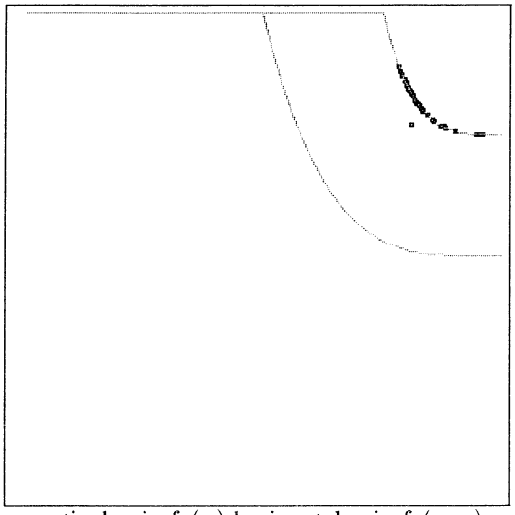


Figure 1: The dynamic pareto optimum GA



vertical axis $f_1(x)$ horizontal axis $f_2(x, y)$

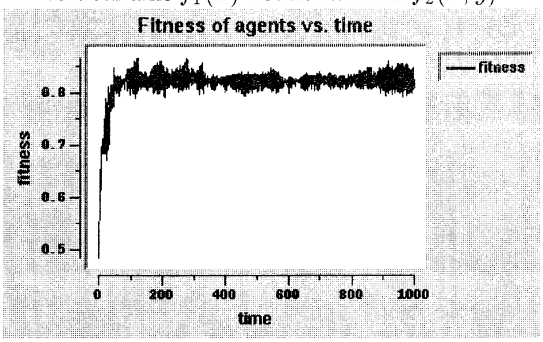


Figure 2: The dynamic pareto optimum GA (seek stability)

3. Do the item 1 and item 2 through all agent pairs who live more than 3 generations.

4. Rank all agents, by the number of agents who dominate agent A as the rank of agent A

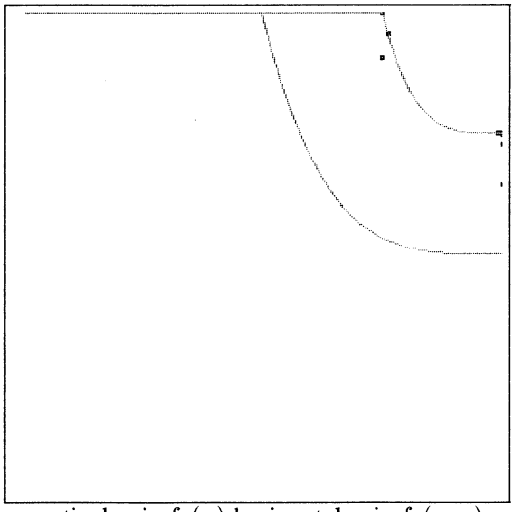
Sharing by distance in the adaptation space is done by following functions.
$d_{ij}^2 = (f_1(x_i) - f_1(x_j))^2 + (f_2(x_i, y_i) - f_2(x_j, y_j))^2$
$s_{ij} = 1.0 - (d_{ij}/30)^{0.2}$
$rankNew_i = rank_i / \sum_j s_{ij}$

Preferences are chosen by one of the next methods.

1. Do not chose the preference.

2. Use the variance of individual time series, correct the ranking as follows.
   $rankNew_i = rank_i/(1 + 10variance_i)$

3. Use the maximum value of individual time series, correct the ranking as follows.
   $rankNew_i = rank_i(1 + 10max_i)$



vertical axis $f_1(x)$ horizontal axis $f_2(x, y)$

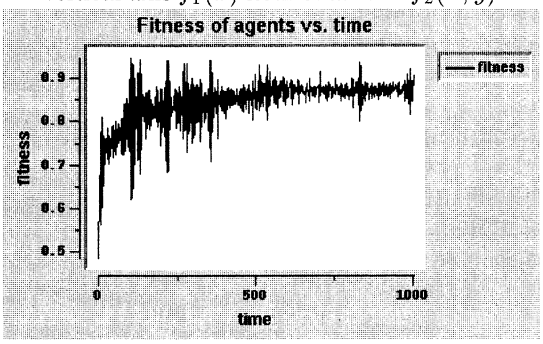


Figure 3: The dynamic pareto optimum GA (seek height)

2.2 Environments

A environment

$f_1(x) = 1.0 - x/100$

B environment

$0 < y < 40 \quad u = (y/100 - 0.2)/0.02$

$g = 4 - 3\exp(-u^2)$

else $\quad u = (y/100 - 0.7)/0.2$

$g = 4 - 2\exp(-u^2)$

$h = max[0, 1.0 - (f_1(x)/g)^4]$

$f_2(x, y) = 1.0 - g * h/4.0$

$f_1(x)$ and $f_2(x, y)$ appear alternately in every generation.

3 experiments

To compare the results, I do the experiments using the next 5 populations.

1. **the dynamic pareto optimum GA** :The dynamic pareto optimum GA without preference
2. **seek stability**:The dynamic pareto optimum GA with preference, which corrects the ranking with variance.
3. **seek height**:The dynamic pareto optimum GA with preference, which corrects the ranking with maximum value.
4. **simple GA**:General GA with selection method in proportion to the fitness value without scaling.
5. **GA ranking**:General GA with Ranking method.

In this time, I show the results of offline performance. Using 100 agents, I do the experiments of 1000 generations.
7 figures of Fig.1,Fig.2,Fig.3,Fig.4,Fig.5 show the agents in the function space.(Dots describe agents.) The right upper curve shows the group of pareto optimum solutions and the another curve shows the group of pseudo pareto optimum solutions.
Fig.1 shows the result of [the dynamic pareto optimum GA] population. Almost agents are located on the pareto optimum line, keeping some distance each other.
Fig.2 shows the result of [seek stability] population. In this case, agents do not keep some distance each other, and they are concentrated near the intersection of the pareto optimum line and diagonal line, where stable solutions are gotten under both $f_1(x)$ function and $f_2(x, y)$ function.
Fig.3 shows the result of [seek height] population. We can see many agents at both sides of the pareto optimum line, where they can get high fitness under either $f_1(x)$ function or $f_2(x, y)$ function.
Fig.4 shows the result of [simple GA] population. The populations of some runs with different seeds fall into pseudo optimum point, but they show totally stable behavior.
Fig.5 shows the result of [GA ranking] population. Behaviors are unstable. Sometimes, the final optimum point is different by seeds. Other time, the population floats on the pareto optimum line and moves.

4 conclution

Recently developed evolutional computation always bases contemporary fitness function.When fitness function changes, the agents in the environments can not compare this time situation with that time situation. It is impossible to make such algorithm support conservative behavior or short-term insight and so on. And if the environments are changed before convergence of population, the instability appears under usual selection method which has relative values among individuals. The reason of these results is The Present Theory Lacks the Evaluation Standard through the Time. Then I introduce pareto optimum which is used in the field of multi-objective optimization into the dynamic problem, make it possible to express conservative behavior or short-term insight and so on. The proposed method (The Dynamic Pareto Optimum GA) makes it possible to put preference into adaptation.
This method is different from other method using memory in the next points. As far as I know, all other works keep old genes or schemas or chromosomes. But this method keeps old fitness values in each individual memory as time series, which make it possible to construct the algorithm with preference of how to adapt.

Bibliography

[1] De Jong,K. Evolving in a Changing World:Springer. 1609 Lecture Note In Artificial Intelligence pp.512-519

[2] Branke,J. Evolutionary Approaches to Dynamic Optimization Problems:A survay. In Evolutionary algorithms for dynamic optimization problems(pp.134-137)

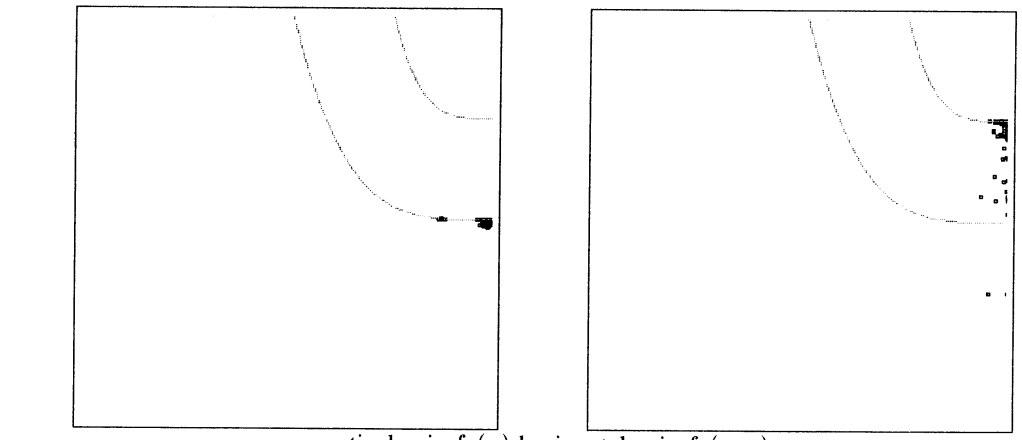
[3] Branke,J. Memory Enhanced Evolutionary Algorithms for Changing OptimizationProblem: CEC 1999,pp.1875-1882

[4] Grefenstette,J.J. Evolvability in Dynamic Landscapes: A Genetic Algorithm Approach:CEC 1999,pp.2031-2037

[5] Deb,K.:Muti-objective Genetic Algorithms:Problem Difficulties and Construction of Test Problems:Evolutionary Computation Vol.7,No.3,pp.205-230,(1999):

[6] Yamasaki,K: The Adaptation Under Dynamic Environments has the Aspect of Multi-Objective Optimization: On line Proceedings of the 9th Workshop of Multi-Agent and Cooperative Computation: (2000)

[7] Yamasaki,K: The problem of Adaptation against Dynamic Environments should include Two Setting: The Proceedings of 13th Symposium of Autonomous Decentralized System: (2001)



vertical axis $f_1(x)$ horizontal axis $f_2(x, y)$

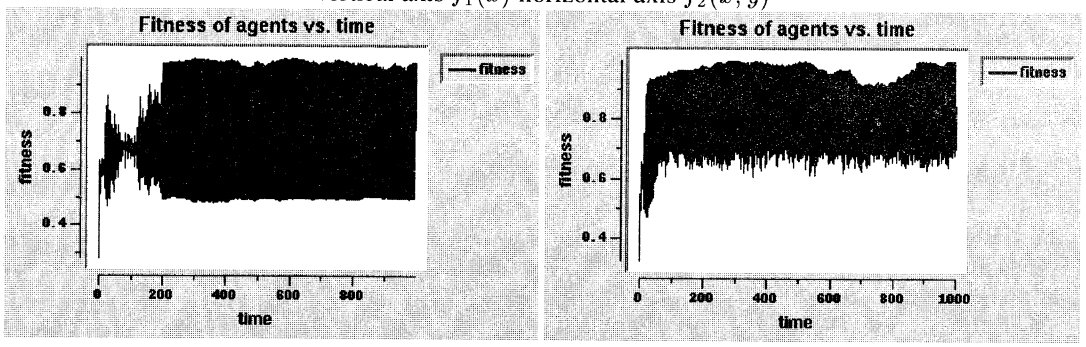
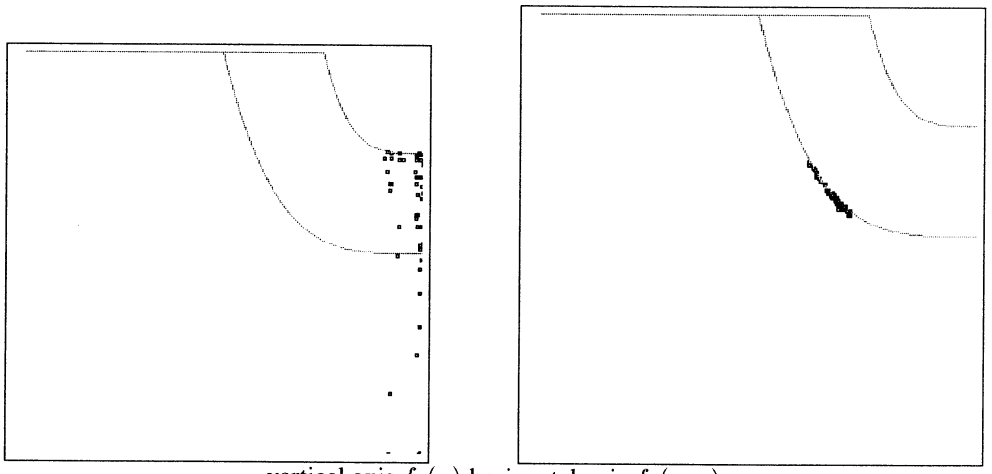


Figure 4: Symple GA



vertical axis $f_1(x)$ horizontal axis $f_2(x, y)$

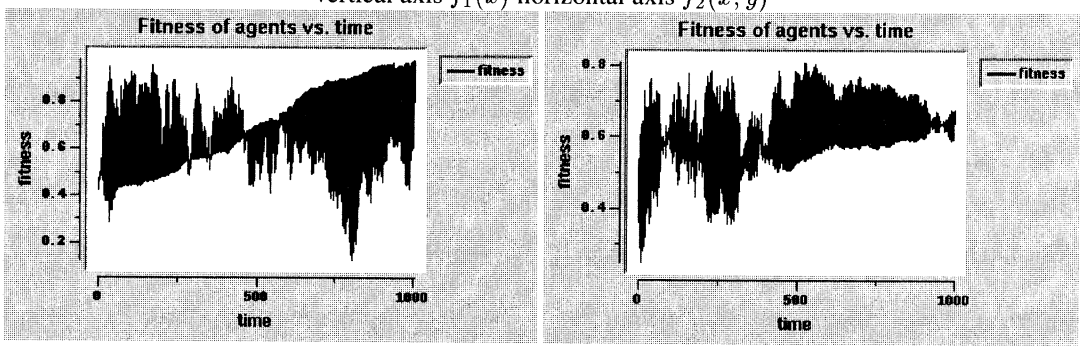


Figure 5: Ranking Selection (relative fitness causes instability)

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Positive Significance of Non-Distributive Law in Real Parallel Processing

Yukio-Pegio Gunji, Hideki Higashi & Masashi Aono

Department of Earth & Planetary Sciences, Faculty of Science;
Graduate School of Science & Technology, Kobe University, Nada Kobe 657 JAPAN

Correspondence to :yg@scipx.planet.kobe-u.ac.jp

How can one express real reciprocal parallel interactions? Imagine that two agents A and B interact with each other. If time is described in a discrete fashion and states of A and B at tth step are expressed as $a(t)$ and $b(t)$, then such an interaction is normally expressed as the simultaneous equations such as $a(t+1) = f^*(a(t), b(t))$ and $b(t+1) = g^*(a(t), b(t))$. The symbol * represents the operation of application of an operator. If one introduces another symbol representing parallel operation by +, one can find that a distributive law results in the simultaneous equations; $(a(t+1), b(t+1)) = (f+g)^*(a(t), b(t)) = f^*(a(t), b(t)) + g^*(a(t), b(t))$. From this, one can see that simultaneous equations are based on the assumption of which an observer can identify any states completely and then he can copy them. This is significance of distributive law. An expression $(a(t), b(t))$ appears twice in simultaneous equations, because of the ability of copy.

In real parallel interactions, there is no time to identify states completely and to copy them. Time never waits for completion of identification. Therefore, while the one agent A identifies the state of the other agent B, the state of B is perpetually changed. As a result, identification is always destined to be incomplete, and distributive law does not hold. The next question arises, can we manifest the positive significance of negative statement (non-distributive law)?

In order to represent the positive significance of non-distributive law, we introduce the formal framework in which some limits are weakened. We construct negative statements (non-distributive law) in weakened (negative) formal system, so that non-distributive law can carry positive significance of parallelism. According to a category theory existence of limits is equivalent to existence of adjunctive functors (MacLane, 1971), and then the weakening adjunction can lead to weakened some limits.

We here take on a concept lattice (Ganter & Wille, 1999), and weaken the adjunction involved in a concept lattice. Concept lattice is based on a formal context that is a triplet, $K(G, M, I)$; a set of objects G, a set of attributes M, and binary relation $I \subseteq G \times M$. Given a formal context, formal concept is defined by a pair (A, B) with $A \subseteq G$ and $B \subseteq M$ such that $SA=B$ and $TB=A$, where

$$SA = \{m \in M \mid gIm, \ ^{\forall}g \in A\} \quad (1)$$
$$TB = \{g \in G \mid gIm, \ ^{\forall}m \in B\}. \quad (2)$$

From this definition, one can see a pair of adjunctive functors $S:P(G) \rightarrow P(M)$ and $T:P(M) \rightarrow P(G)$, because there is one-to-one relationship such that $A \subseteq TB \Leftrightarrow SA \subseteq B$. Here $P(X)$ represents the power set of X.

Because $S:P(G) \rightarrow P(M)$ and $T:P(M) \rightarrow P(G)$ is based on a quantifier $\forall$, the notion of concept is weakened by the weakening of $\forall$. An arbitrary quantifier is the limit of supremum. In stead of $\forall$, we define a weak quantifier by $\forall_p$ as follows;

$$\{^{\forall p}g \in A\} = A \cap ((SA)^c)^+ \quad (3)$$

where X^c represents complement of X and

$$B^+ = \{g \in G \mid gIm, \ ^{\exists}m \in B\}. \quad (4)$$

Eq.(3) implies the notion of wholeness dependent on an object A. Compared with

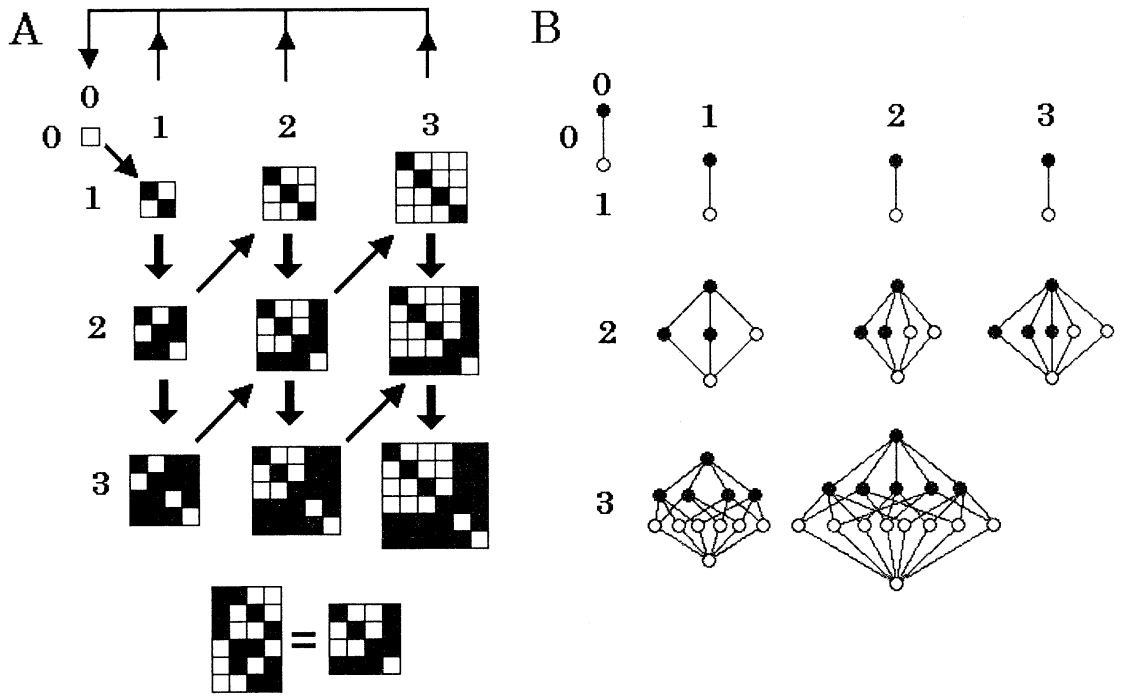


Fig. 1 A. Dynamical change of a formal context derived from the context based on Eqs. (5) and (6). A set of black squares represents I_t of $K(G_t, M_t, I_t)$. In the formal concept lattice theory (Ganter & Wille, 1999), contexts in the same column represent the compatible contexts (i.e., The corresponding formal concept lattices are isomorphic to one another), because the transformation denoted by thick arrows represents compatible transformation of a context. In contrast, in formal concepts based on $FA=B$ and $UB=A$ (see text) this compatible transformation represents the weakening distributive law. Thin arrows represent the transformation from $K(G_t, M_t, I_t)$ to $K(G_{t+1}, M_{t+1}, I_{t+1})$. B. A formal concept lattice corresponding to each formal context in the same coordinate. At the upper row, Boolean lattices are located. At the middle row, non-distributive modular lattices are located. At the lower low, non-modular ortho-modular lattices are located. Toward the bottom, distributive law is weakened. If the semantics of computation is based on a dynamical lattice, then the change from deterministic computations to stochastic ones is possible thanks to the change of the domain in which distributive law holds.

normal arbitrary quantifier, weak quantifier carries the negative expression in the form of $((SA)^c)^+$. This carries ambiguous meaning of both an arbitrary quantifier and the wholeness in the diagonal argument (Gunji et al., 2000; 2001).

By introducing a weak quantifier, dynamical formal context can be made. Formal context is defined as a triplet of $K(G_t, M_t, I_t)$, and formal context (A, B) with $A \subseteq G_t$, and $B \subseteq M_{t+1}$, such that $FA=B$ and $UB=A$, where

$$FA = \{m \in M_t, \mid gI_t m, \ ^{\forall p} g \in A\} \quad (5)$$
$$UB = \{g \in G_{t+1} \mid gI_{t+1} m, \ ^{\forall} m \in B\}. \quad (6)$$

Actually, given $K(G_t, M_t, I_t)$, for any $A \subseteq G_t$, FA is defined. After that, so as to satisfy Eq.(6), G_{t+1} and I_{t+1} are constructed as follows; First, for the greatest element M, $\emptyset$ is defined for $UB=A$, and then a concept $(\emptyset, M)$ is obtained. For $B_1 \subseteq B_2$ (i.e., $FA_1 \subseteq FA_2$), new extent is made to satisfy $A_2 \supseteq A_1$. If $B_2 \subseteq B_1$ and $B_3 \subseteq B_1$, it is defined that $A_2 = A_1 \cup \{*1\}$ and $A_3 = A_1 \cup \{*2\}$ with $\{*1\} \neq \{*2\}$, where $\{*k\}$ represents a singleton set. If $B_3 \subseteq B_1$ and $B_3 \subseteq B_2$, it is defined that $A_3 = A_1 \cup A_2$.

As a result, a concept lattice consists of $(UB_1, FA_1) \leqq (UB_2, FA_2)$ with $UB_1 \subseteq UB_2$ and $FA_1 \supseteq FA_2$. This concept lattice is denoted by $B(G_{t+1}, M_{t+1}, I_{t+1})$, and it leads

to the transformation of a formal context from $K(G_t, M_t, I_t)$ to $K(G_{t+1}, M_{t+1}, I_{t+1})$.

The dynamical change of a concept lattice is used as a semantics of a transition rule expressed by a particular lattice polynomial, $f(a_{k-1}^t, a_k^t, a_{k+1}^t)$. Given a transition rule such as

$$a_k^{t+1} = \nu(f(a_{k-1}^t, a_k^t, a_{k+1}^t)), \qquad (7)$$

and an initial binary sequence, $b^0=(a_0^t, a_1^t, .., a_N^t)$, assume that b^1 is generated by applying Eq.(7) in a synchronous updating with periodic boundary condition. A state of a_k^{t+1} is either 0 or 1, and is obtained by ν: $B(G_t, M_t, I_t)\rightarrow\{0,1\}$. A lattice polynomial, $f(a_{k-1}^t, a_k^t, a_{k+1}^t)$ is calculated in $B(G_t, M_t, I_t)$, and elements of the lattice is first mapped into $B(G_t, M_t, I_t)$, and then is mapped to $\{0,1\}$. The compatible transformation of a context, C, is defined by

$$C(K(G_t, M_t, I_t)) = K(G^1_t, M^1_t, I^1_t) = K(G_t \cup \{h\}, \quad M_t \cup \{n\}, \quad I_t \cup (\{h\} \times M_t) \cup (G_t \times \{n\})), \qquad (8)$$

and $C^n(K(G_t, M_t, I_t)) = K(G^n_t, M^n_t, I^n_t)$ is also defined by $K(G^{n-1}_t \cup \{h\}, \quad M^{n-1}_t \cup \{n\}, \quad I^{n-1}_t \cup (\{h\} \times M^{n-1}_t) \cup (G^{n-1}_t \times \{n\}))$. The operation $\gamma: N \rightarrow N$ is defined by

$$\gamma(b^t) = \left(\sum_{k=1}^{N} a_k^t\right) \bmod 3. \qquad (9)$$

Time transition of a binary sequence based on a dynamical lattice is defined as follows; First, to satisfy ν: $B(G_t, M_t, I_t)\rightarrow\{0,1\}$, an element of binary sequence a_k^t, is mapped into an element of a lattice, $B(G_t, M_t, I_t)$. Second, $C^{\gamma(b^t)}(K(G_t, M_t, I_t))$ is calculated. From this context, by using Eqs. (5) and (6), a new lattice $B(G_{t+1}, M_{t+1}, I_{t+1})$ and context $K(G_{t+1}, M_{t+1}, I_{t+1})$ is obtained. In this lattice, Eq.(7) is calculated, and b^t is finally obtained by applying ν.

In starting from $K(G_0, M_0, I_0)$ with $G_0=\{*\}$, $M_0=\{*\}$ and $I_0=\emptyset$, $B(G_0, M_0, I_0)$ is a Boolean (complemented distributed) lattice. Depending on b^t, $B(G_1, M_1, I_1)$ is changed with respect to the size of domain in which the distributive law holds. If $\gamma(b^t)=0$, 1 and 2, then $B(G_1, M_1, I_1)$ is Boolean (distributive), non-distributive modular, non-modular ortho-modular lattice, respectively. There is a hierarchical class as shown in Fig. 1, and $K(G_1, M_1, I_1)$ calculated from $B(G_1, M_1, I_1)$ is lower class than $K(G_0, M_0, I_0)$ with respect to distributive law.

Complement of an element in non-distributive lattice is not uniquely determined. Therefore, dynamical change of a lattice with respect to distributive law can contain the drastic change from deterministic computations to stochastic ones. Because the dynamic change of a lattice results both from compatible transformation of contexts leading to a less distributive lattice and from weak adjunction leading to a more distributive lattice, non-distributive law can carry positive significance that is a distributive law in future.

In addition, in-deterministic process in choosing a complement can carry the positive significance of non-distributive (negative) law by previous states. Because an element of binary sequence b^t is determined as an element of a lattice $B(G_t, M_t, I_t)$ to satisfy ν: $B(G_t, M_t, I_t)\rightarrow\{0,1\}$, a_k^t, can carry the previous state of either 0 and 1. Therefore this pervious state can be used as the help of choosing complement in a non-distributive lattice. As a result, even a stochastic computation can carry the positive significance not in the from of randomness, and leads to a particular pattern that differs from just a disorder.

We focus on weakened formal framework in which non-distributive law can carry the positive significance. For this aim, first we take a formal concept lattice based on the adjunctive functors, and then weaken the adjunction by weakening an arbitrary quantifier. Because an arbitrary quantifier is a kind of limit of supremum, weakened arbitrary quantifier can lead to the modification of operations defined in a lattice, such as complement. As a result, choice of a complement in a non-distributive lattice has the positive significance of negative expression, and that leads to a particular order in stochastic computation.

Autistics' ability yields one of the most intriguing phenomena containing the positive significance of parallel processing. Some of autistics can grasp the number of coins distributed on a table at a glance. It may be possible because they can watch many coins in a parallel fashion. In other words they can use the positive significance

of parallelism per se.

Happé (1996, 1999) explains the ability of autistics by weakening central coherence. Normal peoples succumb visual illusions because they re-interpret the meaning of a part in a whole diagram owing to the central coherence. In contrasts, autistics do not succumb the image of wholeness, they can correctly identify the size or length of a particular part in a whole diagram. By the weakening central coherence, some abilities of autistics (e.g., no succumbing visual illusions) can be explained, but some cannot be. In grasping the number of distributed coins, they can reduce the number from a visual pattern and can tell the number to other people. Therefore, there are another kind of central coherence in autistics.

The notion of weakening, per se, can carry the negative impression of negative expression. Another kind of central coherence results from the positive significance of negative (non-distributive) expressions of parallel processing. In our framework, the ability of autistics and/or the positive significance of real parallel interactions may be explained and can be evoked.

References

Ganter, B. & Wille, R. 1999, *Formal Concept Analysis*. Springer, Berlin.

Gunji, P.-Y., Kusunoki, Y. & Aono, M. 2000, Interface of global and local semantics in a self-navigating system based on the concept lattice. *Chaos, Solitons & Fractals* (in press).

Gunji, P.-Y., Aono, M. Higashi, H. & Takachi, Y. 2001, The third wholeness as an endo-observer. *In: Science of Interface* (Diebner, H. & Rössler, O.E. eds.), MIT Press (in press).

Happé, F. 1996, Autism: cognitive deficit or cognitive style? *Trends Cogn.Sci.* 3, 216-222.

Happé, F. 1999, Studying weak central coherence at low levels: children with autism do not succumb to visual illusion: a research note. *J. Children Psychol. Psychiatry* 3, 873-877.

MacLane, S. 1971, *Categories for Working Mathematicians*. Springer, Berlin.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Muu: Embodied Interface for Social Bonding

SAKAMOTO Shoji SUZUKI Noriko OKADA Michio
ATR Media Integration & Communications Research Laboratories
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288 Japan

Abstract

The recent spread of the Internet has provided access to a variety of information sources and enriched our ways of communication. On the other hand, it has been pointed out that the "digital divide" between the haves and have-nots of access to information is becoming a serious problem in our society.

In this paper, we introduce our life-like creature as a new type of interface to computer systems. Our creature, Muu, displays social cues while speaking with vocal sounds. Human participants can enjoy having a chat with them by simply talking to them, and no special interaction skills are required.

We have implemented a system to perform echoic mimicry using inarticulate sounds. From observation at SIGGRAPH 2000 exhibition, attendees were observed to chat with Muu successfully by only using ordinary communication skills.

1 Introduction

The primary motivation of our everyday conversation is not always to convey messages to others but rather to confirm social bonding, coordinate social bonding, and coordinate social distance during the interaction. From this viewpoint, we are rethinking the social aspects of artificial creatures and the social skills used to establish social bonding with humans in order to devise social entities in human-inhabited environments.

"Muu" is our artificial creature with a humorous appearance of a simple physical body. The challenge here is to develop an artifact that can establish social bonding with a human. The creatures also works as an embodied interface that mediates the social bonding people have established in their everyday conversations. The mechanism used to create social bonding is implemented with the Learning Classifier System, which features a self-referencing that coordinates the relationship with a human during an interaction.

2 Coordination in Everyday Chat

2.1 Peer Relationships

According to the traditional code model, communications can be regarded as conveying messages between sender and receiver. This model is useful to describe networked computers. Human users, however, must master certain skills to encode and decode messages in order to interact with computers. It could be frustrating for children and senior citizens to keep watching computer screens and searching for the '~' key on a keyboard.

On the other hand, no special communication skill is required in our everyday communication. Everyday chat is a natural form of communication, as opposed contrary to the code model type of communication, since it is a proto-goal type of communication rather than a task-oriented dialogue[7]. In everyday chat, there is no overall scenario, no sender of commands and no receiver to follow the instructions. All of the participants are in peer relationships, and their purpose is chatting itself rather than achieving a constructive conclusion. A good example is the long phone calls of school girls. They do not have a lot of messages to convey but seem to be confirming that they are bonded.

2.2 Communication as Forming Relationships

The purpose of everyday chat is to confirm that the participants are socially bonded, not to exchange profitable messages. This type of conversation is different from task-oriented dialogues because there is no goal to achieve nor scenario used to aquire the pieces of information to fill vacant slots. This implies that the traditional techniques of dialogue management cannot be easily applied to model this phenomina.

In 'Talking Eye' project[3], we have introduced a new idea to describe the utterance chain in this type of conversation. We have focused on the following two aspects of an utterance.

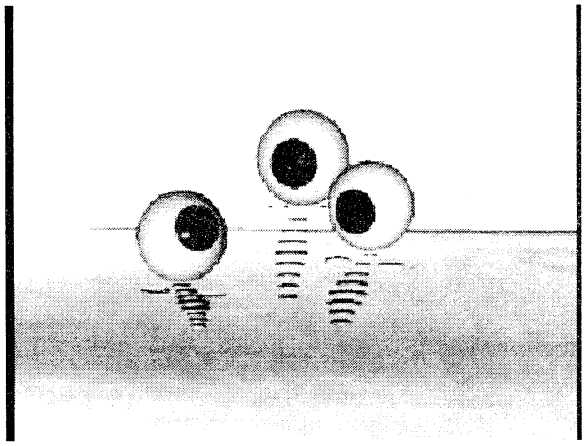


Figure 1: Talking Eye

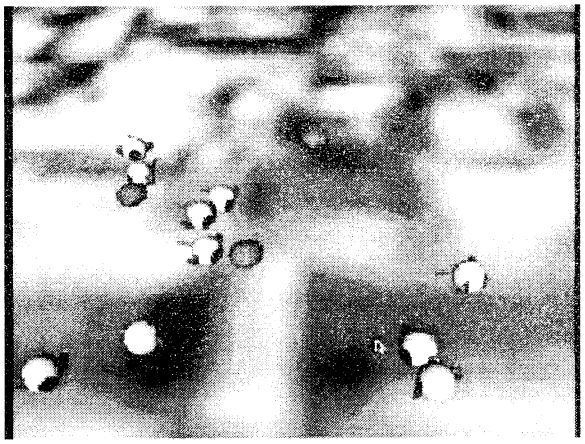


Figure 2: Eye-tadpole

- Entrusting: A spontaneous utterance of a talker involves indeterminacy, such as "Who among the participants would react to this utterance?" or "In what way?". Even with indeterminacy, the talker spontaneously entrusts his utterance in chatting.
- Grounding: A reaction to an entrusted utterance will resolve the indeterminacy. The meaning of an utterance is determined when it is supported by a coressponding utterance.

Since a grounding utterance presents an aspect of entrusting at the same time, it leads to a succeeding utterance and this will result in a chain of utterances. This process enabled the artifact 'Talking Eye'(Fig. 1) to keep chatting without a previously designed scenario of conversation.

As discussed above, the indeterminacy of an utterance cannot be resolved by an individual in isolation. Mutual coordination through interactions would appear to be a key to this problem. To confirm this idea, we built the 'Eye-tadpole' system, which is composed of a group of eyeball-shaped tadpoles and some balls(Fig. 2). The eye-tadpoles behave according to two types of desires: to reach the ball and to be close to others. These two desires are mutually exclusive; for example, when one eye-tadpole keeps touching a ball, it makes other eye-tadpoles frustrated, causing them go away. To find a balance point of the two desires, the behavior of each eye-tadpole is controlled through dual dynamics(Fig. 3). One is called social referencing dynamics, and this is used to control the interactions with the environment to produce the spontaneous behaviors. The other is called self referencing dynamics, which is used to help the individual locate himself in a parameter space of the two types of desires and to design his strategy. Both of the dynamics are implemented as Learning Classifier Systems, and they restrict each other. The self referencing dynamics receives an evaluation value of the invoked schemata in the the social referencing dynamics, which is used to find an appropriate position in the parameter space. The revised position in the parameter space of the self referencing dynamics would take effect in assigning priorities to schemata in the social referencing dynamics, restricting which schema would be evoked.

To keep the eye-tadpoles in good harmony, it is important for each of them to find a social niche by coodinating their behavior with other social entities. Through this coordination, the appropriate relationship with social others is formed, allowing the individual to settle into a niche. From this viewpoint, communication can be reconsidered relationship forming, not merely conveying messages.

3 Embodied Interface

3.1 Muu

The current interface to computer systems cannot adequately apply our natural skills of communication. There seems to be a phsycological barrier to speaking to a CRT screen filled with rectangle windows. This type of interface is designed to perform direct manipulation of computers. This type of interface makes us

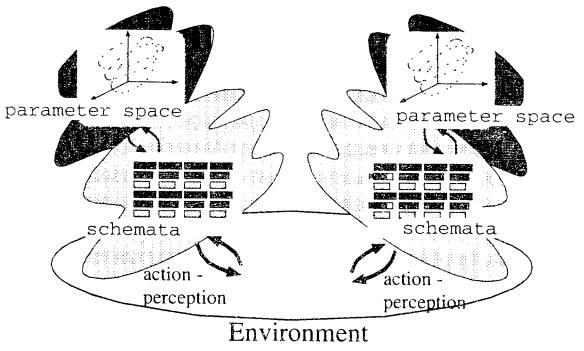


Figure 3: Dual dynamics

perceive the computer as a tool to utilize, not as a partner to communicate with. It has been pointed out that our attitudes toward living creatures are different from those toward inanimate objects[1]. In order to draw out our natural skills of communication, a different type of interface is required.

As the first step of the new interface, we have built a life-like creatures named 'Muu(Fig. 4),' with the folloing features.

- Neutral design.
- Ability to displaying social cues.
- Multi party conversation.

We would expect dog shaped robots to behave like dogs, and we may treat them like pets. To establish a peer relationship with an artifact, this is not adequate. Therefore, we selected a very abstracted shape that does not remind users of any existing animal. On the other hand, we introduced some features generally classified as infant schemata, such as round shape and wide open eyes.

Muu nods or shakes when listening to other Muu, as well as changing the direction of gaze. This basic skill in communication helps to draw out the intentional stance from humans. We have purposely designed Muu not to have functions that present facial expressions. This is meant to present an indeterminacy of expression and to elicit original interpretations of Muu's behavior from humans in peer relationships.

Not simply reactive to events such as being touched or spoken to, Muu's can talk on their own. This presents a field of conversation that a human can join as an additional participant by using natual communication skills.

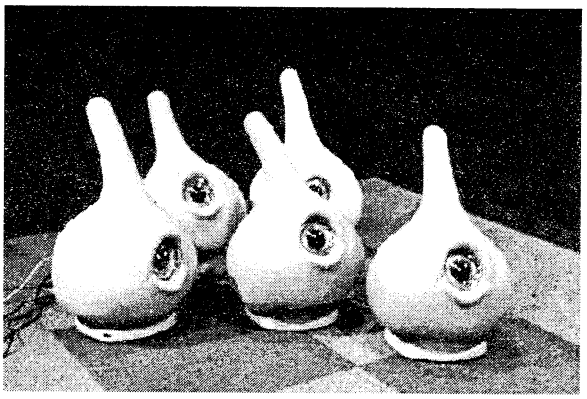


Figure 4: Muu

3.2 Echoic Mimicry Using Inarticulate Sounds

Humans are apt to infer a partner's intention or emotion when the partner mimics their utterance echoicly. This is true for partners who have insufficient linguistic ability such as babies or animals. In this case, humans seem to focus on phonetic features, not on language content, to notice the partner's mimicry. A system to perform echoic mimicry using inarticulate sounds has been developed in our research group. We applied this to Muu to observe if human participants can chat with it.

3.2.1 Implementation

The Learning Classifier System used for the eye-tadpole is composed of a set of flat sequences of schemata. This is adequate to bulid a system that is simply reactive to discrete events but insufficient to describe the behavior of conversation, which is made up of a set of behaviors in a sequence. Therefore, we introduced a group structure of the schemata to describe its scope. Each group is tagged with a time section, and scope transition is performed according to time. This helps to describe the temporal structure of the schemata. Explicit transition is also available to describe the next expected behavior of the system.

A Muu has a pan-tilt video camera and some sensors installed. It is controlled with a networked PC, the network is used as the media for interaction between Muu's(Fig. 5).

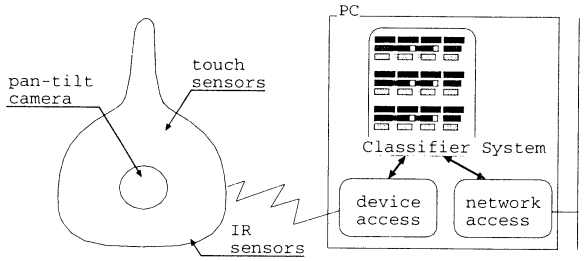


Figure 5: structure of Muu

Figure 6: At SIGGRAPH 2000

3.2.2 Experiment

We exhibited the system at SIGGRAPH 2000[6] and asked the participants to have a chat with it. In a typical case of interaction, a participant first gets on a chair in front of the Muu's, starts waving at them and touching them. Once he notices that the Muu's can speak, he or she picks up the microphone and starts talking to them. Some of the participants made his original interpretations of the Muu's voice and successfully collaborated to have a chat(Fig. 6).

4 Conclusions

In this paper, we pointed out that the traditional interface to computer systems requires specific skills to convey messages and that this can cause a "digital divide" between the haves and have-nots of access skills. This is because the current interface of computers is based on the traditional code model.

To overcome this problem, we conducted a research on artifacts that can make everyday chat. Furthermore, we pointed out that establishing a peer relationship is essential to this proto-goal type of problem. This is also true for a system where behavior is designed based on the coordination of others.

We have proposed the application of life-like creatures as an interface of computers to establish a peer relationship with them. We have implemented a system to perform echoic mimicry using inarticulate sounds. From observations at the SIGGRAPH 2000 exhibition, attendees successfully chat with Muu's by only using ordinary communication skills.

In future work, we plan to add a mechanism for the aquisition of communication skills and to refine the software for a direct expression of the entrusting-grounding model.

References

[1] Dennett D.C., *Kinds of minds*. HarperCollins Publishres, 1996.

[2] Suchman L.A., *Plans and Situated Actions*. Cambridge University Press, 1987.

[3] Suzuki N., Takeuchi Y., Ishii K., Okada M., "Talking Eye: Autonomous creatures for augmented chatting", *Robotics and Autonomous Systems*, Vol. 31, pp. 171-184, 2000.

[4] Shiose T., Sawaragi T., Katai O., Okada M., "Segregated Sense of Values within Multiagent Systems through Reciprocity Enabled by a Bi-Referential Model", *Australia Japan Joint Workshop on Intelligent and Evolutionary Systems*, F3-4, pp. 107–114, 1998.

[5] Suzuki N., Takeuchi Y., Ishii K., Okada M., "Evaluation of Affiliation in Iteraction with Autonomous Creatures", *Proc. of EuroSpeech-99*, S2.OR3.5, pp. 259–262, 1999.

[6] Okada M., "Muu: Artificial Creatures as an Embodied Interface", *Proc. of SIGGRAPH*, 2000.

[7] Seto S., Nagata Y., Kanazawa H., "Spontaneous speech dialogue system TOSBURG II and its evaluation", *Proc. of ISSD-93*, pp. 41-44, 1993.

Proc. of The Sixth Int. Symp. on Artificial Life and Robotics (AROB 6th' 01)
Tokyo Japan, 15-17, January, 2001,

Computation and Life in a Reversible Cellular Space

Kenichi MORITA, and Katsunobu IMAI
Faculty of Engineering, Hiroshima University
Higashi-Hiroshima, 739-8527, Japan
{morita, imai}@ke.sys.hiroshima-u.ac.jp

Abstract

A reversible cellular automaton (RCA) is a "backward deterministic" CA in which every configuration of a cellular space has at most one predecessor. It is an abstract model of a physically reversible space. In spite of the strong constraint of reversiblity, they have very rich ability of computing, and various interesting life-like phenomena can emerge in such a cellular space. We first discuss computation-universality of RCAs, and show several simple 2-D RCA models having this property. Next we discuss the problem of self-reproduction in RCAs. We describe our previous model of 2-D RCA in which a variety of objects can reproduce themselves by "shape-encoding" mechanism. We also show our recent RCA model of 3-D self-reproduction.

Keywords: self-reproduction, cellular automata, reversible computing, computation-universality

1 Introduction

A reversible cellular automaton (RCA) is an artificial and abstract model of space and time reflecting physical reversibility. Since our physical world is quantum mechanical (in a microscopic level) and its evolution is reversible, it is reasonable and important to investigate RCAs and their information processing abilities as well as related spatiotemporal phenomena. On the other hand, it is well known that von Neumann [1] first designed a self-reproducing CA, i.e., he showed it is possible to construct a universal computer (Turing machine) that can reproduce itself in a 29-state 2-D (irreversible) cellular space. In this paper, we investigate and discuss the problems of how the functions of computing and self-reproduction can be realized in a reversible environment, especially in a reversible cellular space. As we shall see, in spite of the strong constraint of reversibility, RCAs have rich abilities of computing and self-reproduction.

As for the computing ability of RCAs, Toffoli [2] first showed that a 2-D RCA is computation-universal. Later, Morita and Harao [3] strengthened his result by showing that a 1-D RCA is universal. For the case of 2-D RCA, Margolus [4] proposed an interesting 2-state model with so-called Margolus neighborhood which is a little different framework from the usual CA. After that, several simple universal models of 2-D RCAs using the framework of a partitioned CA (PCA) have been given [5, 6, 7, 8]. From these results we can see that universal computers can be constructed based on very simple and primitive reversible rules. Furthermore, in some cases, computation in RCAs can be carried out in a very different way from that of conventional computers. Thus it may lead new insights for future architectures of computing. Here, we discuss how universal computing mechanisms can be decomposed into such simple reversible rules.

Self-reproduction is also possible in an RCA. At present, we assume the so-called Langton's criterion [9], i.e., self-reproducing objects need not have computation-universality, but the construction of a daughter object should be actively directed by its mother by using its "gene" properly. We first show our previous model of 2-D self-reproducing RCA called SR_8 [10], where various shapes of objects called Worms and Loops can self-reproduce. Such objects have abilities of (1) encoding the shape into a "gene" represented by a command sequence (it is called a "shape-encoding" mechanism; see also [11]), (2) copying the gene, and (3) interpreting the gene to create a shape. We can design so that these operations are all performed reversibly. Furthermore, because of the shape-encoding mechanism, a Worm or a Loop of almost any shape can self-reproduce in a very simple fashion. We also describe our recent attempt to create a 3-D self-reproducing RCA (SR_9) [12]. Its basic mechanism is similar to that of the 2-D case. However, it is not a mere extension of the 2-D model, because varieties of possible shapes and arrangements of Worms and Loops are much greater than that of the 2-D case.

2 Computation in RCAs

2.1 A partitioned CA (PCA)

Here, we use a framework of a partitioned CA (PCA) [3], which is a subclass of usual CAs. It greatly helps to design an RCA, because, in a PCA, injectivity (reversibility) of a global map is equivalent to injectivity of its local map [3]. A 2-D PCA with von Neumann neighborhood is a one whose cell is divided into five parts as shown in Fig.1.

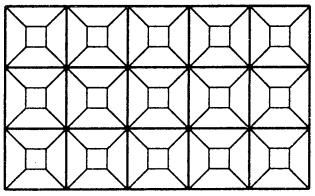


Figure 1: Cellular space of a 5-neighbor PCA.

The next state of each cell is determined by the present states of the center part of this cell, the lower part of the upper cell, the left part of the right cell, the upper part of the lower cell, and the right part of the left cell (not depending on the entire five cells). Hence, each transition rule is depicted as in Fig.2.

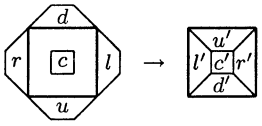


Figure 2: A rule of a 5-neighbor PCA.

Note that, in the case that the center part has only one state (thus it is actually the 4-neighbor case), the center part is omitted in a figure.

2.2 Simple universal 2-D RPCAs

If we want to construct a computation-universal *irreversible* 2-D CA, it is sufficient to construct a one such that conventional logic gates AND, OR, and NOT as well as routing of signals can be realized in its cellular space. However, since AND and OR are irreversible gates (in the sense that inputs are not uniquely determined from its output), it is not possible to design an RCA that directly realizes these gates. Instead, we take another approach, i.e., we construct RCAs that can embed a universal reversible logic element.

A Fredkin gate (F-gate) [13] is a reversible (i.e., its logical function is one-to-one) and bit-conserving (i.e., the number of 1's is conserved between inputs and outputs) logic gate shown in Fig.3. It is known as a universal reversible gate [13].

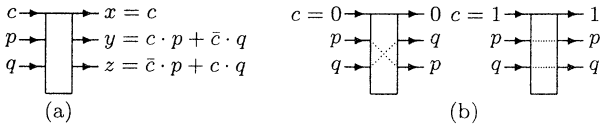


Figure 3: (a) A Fredkin gate, and (b) its function.

Two universal models of 4-neighbour 2^4-state RPCAs (called S_1 and S_2), in which F-gate can be embedded, have been given in [6]. Fig.4 shows the set of transition rules of S_1 (note that since S_1 is rotation-symmetric, rotated rules are omitted). Fig.5 shows a realization of a switch-gate (S-gate), a 2-input 3-output reversible gate. Since an F-gate can be constructed by switch-gates [13], S_1 realizes an F-gate.

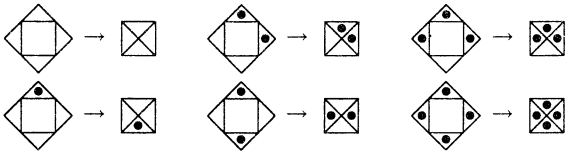


Figure 4: Transition rules of the universal RPCA S_1.

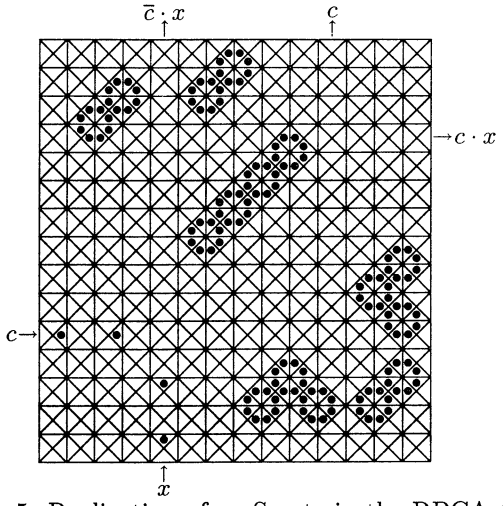


Figure 5: Realization of an S-gate in the RPCA S_1.

It is also possible to construct RPCAs having other types of grids. Imai and Morita [5] gave a universal 3-neighbor 2^3-state "triangular" RPCA T_1. T_1 has extremely simple transition rules as shown in Fig.6, and can realize an F-gate.

Besides an F-gate, a reversible logic element called a "rotary element" (Fig.7) is also known to be universal [7, 14]. It has been shown that a 4-neigbor 4^4-state RPCA model P_4 [6] and a 3^4-state model P_3 [8] can embed a rotary element. In these models, any reversible counter machines can be embedded very concisely. Furthermore, its design method is very different from the conventional one.

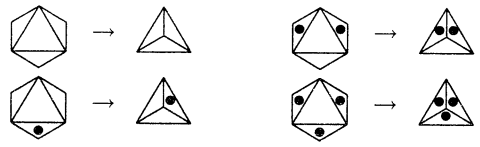


Figure 6: Transition rules of the triangular RPCA T_1.

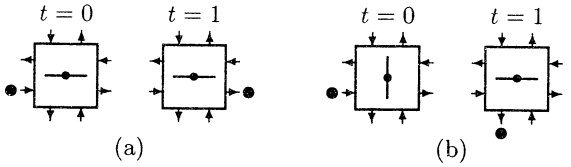


Figure 7: Operations of a rotary element: (a) the parallel case (the coming direction of a particle is parallel to the rotating bar), and (b) the orthogonal case.

3 Self-Reproduction in RPCAs

3.1 2-D self-reproduction

A 2-D 5-neighbor self-reproducing RPCA called SR_8 has been proposed in [10]. Each cell of SR_8 has 8^5 states (i.e., each of five parts of a cell has 8-state).

As in the 29-state CA of von Neumann, SR_8 also makes use of a genetic code (description of the object's shape). That is, the body of a daughter object is constructed by interpreting the description. Furthermore, if the machine can encode its shape into a description by checking its body dynamically, there is no need to keep the entire description. In fact, there have been a few models that performs self-reproduction in such a manner [10, 11, 15, 16]. The method employed in SR_8 and a 2-D 12-state (irreversible) CA model SR_{12} [11] is called a "shape-encoding" mechanism. This method has the following advantages:

1. Complexity of a self-reproducing configuration is very low. For example, a Worm or a Loop with only 4 cells can self-reproduce.
2. A Worm or a Loop of an arbitrary shape can self-reproduce. Thus, self-reproducing ability of an object is relatively robust against distortion.
3. Since shape-encoding and decoding are performed directly and symmetrically, their mechanisms can be easily understood.

Furthermore, in this cellular space, three main operations (i.e., encoding the shape of an object into a gene represented by a command sequence, copying the gene, and interpreting the gene to create an object) are all performed *reversibly*.

A *Worm* is a simple signal wire with two open ends: a head and a tail. At the tail cell the shape of the Worm is "encoded" into commands and the tail retracts one by one. The commands are sent to the head along the wire. At the head of the Worm, commands are decoded and executed to extend the head. Therefore, it crawls in the space keeping its shape cyclically. By putting a "branch" command, which makes a head branch, a Worm can self-reproduce as in Fig.8.

A *Loop* is a simple closed signal wire. It can also reproduce itself in a similar way as in a Worm by extending a "constructing arm" as shown in Fig.9.

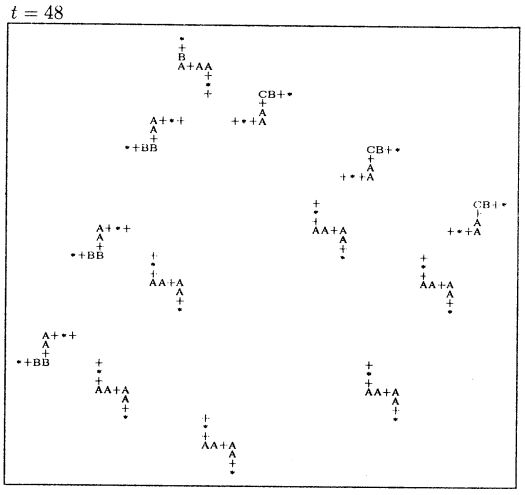


Figure 8: A self-reproducing Worm in SR_8.

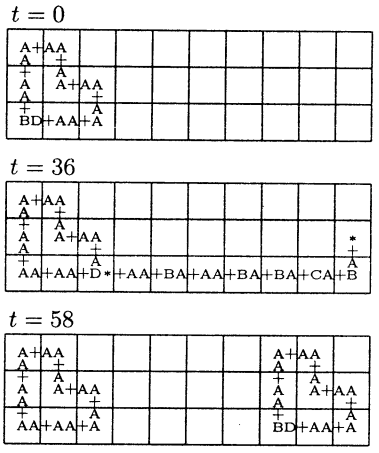


Figure 9: A self-reproducing Loop in SR_8.

3.2 3-D self-reproduction

By extending SR_8, we gave a 3-D self-reproducing RPCA SR_9 [12]. This model is a 7-neighbor RPCA, and has 9 states in each of seven parts of a cell (hence each cell has 9^7 states). As in SR_8, Worms and Loops of various shapes can reproduce themselves in SR_9.

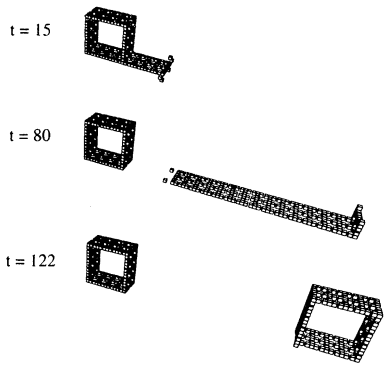


Figure 10: Self-reproduction of a 3-D Loop.

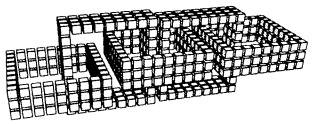


Figure 11: A chain formed by a self-reproducing Loop.

In the 3-D model, varieties of possible shapes and arrangements of Worms and Loops are much greater than that of 2-D. Fig.10 shows a simple self-reproduction process of a 3-D Loop. By controlling a position of a constructing arm by a command sequence, we can design a Loop such that it produces a semi-infinite chain as in Fig.11. Another example of a Loop that forms a pile of Loops is shown in Fig.12.

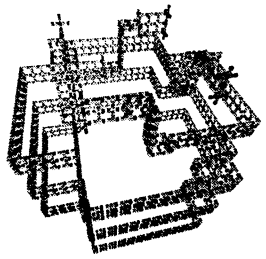


Figure 12: A more complex self-reproducing Loop.

4 Concluding Remarks

We have investigated how computing and self-reproduction tasks are performed in RCAs, and found that these tasks are decomposed into very primitive reversible rules. However, there remains a problem to construct an RCA having both computing and self-reproduction abilities in an elegant way.

Acknowledgements: This work was supported in part by Grant-in-Aid for Scientific Research (C) No. 12680353 from Ministry of Education, Science, Sports and Culture of Japan, and by the Okawa Foundation for Information and Telecommunication.

References

[1] von Neumann, J., *Theory of Self-reproducing Automata* (ed. A.W.Burks), The University of Illinois Press, Urbana (1966).

[2] Toffoli, T., Computation and construction universality of reversible cellular automata, *J. Comput. Syst. Sci.*, **15**, 213–231 (1977).

[3] Morita, K., and Harao, M., Computation universality of one-dimensional reversible (injective) cellular automata, *Trans. IEICE Japan*, **E-72**, 758–762 (1989).

[4] Margolus, N., Physics-like model of computation, *Physica*, **10D**, 81–95 (1984).

[5] Imai, K., and Morita, K., A computation-universal two-dimensional 8-state triangular reversible cellular automaton, *Theoret. Comput. Sci.*, **231**, 181–191 (2000).

[6] Morita, K., and Ueno, S., Computation-universal models of two-dimensional 16-state reversible cellular automata, *IEICE Trans. Inf. & Syst.*, **E75-D**, 141–147 (1992).

[7] Morita, K., Tojima, Y., and Imai, K., A simple computer embedded in a reversible and number-conserving two-dimensional cellular space, *Multiple-Valued Logic*, (to appear) (2000).
http://www.ke.sys.hiroshima-u.ac.jp/~morita/p4/

[8] Morita, K., and Ogiro, T., Embedding a counter machine in a simple reversible 2-D cellular space, *Proc. 6th IFIP Int. Workshop on Cellular Automata*, Osaka, 30–31 (2000).
http://www.ke.sys.hiroshima-u.ac.jp/~morita/p3/

[9] Langton, C.G., Self-reproduction in cellular automata, *Physica*, **10D**, 135–144 (1984).

[10] Morita, K., and Imai, K., Self-reproduction in a reversible cellular space, *Theoret. Comput. Sci.*, **168**, 337–366 (1996).
http://www.ke.sys.hiroshima-u.ac.jp/projects/rca/sr/

[11] Morita, K., and Imai, K., A simple self-reproducing cellular automaton with shape-encoding mechanism, *Artificial Life V* (eds. C.G. Langton and K. Shimohara), The MIT Press, 489–496 (1997).

[12] Hori, T., Imai, K., and Morita, K., Self-reproduction movies of 3D reversible cellular automata (1998).
http://kelp.ke.sys.hiroshima-u.ac.jp/projects/rca/sr3d/

[13] Fredkin, E., and Toffoli, T., Conservative logic, *Int. J. Theoret. Phys.*, **21**, 219–253 (1982).

[14] Morita, K., A new universal logic element for reversible computing, Technical Report of IEICE Japan, COMP99-94 (2000).

[15] Ibáñez, J., Anabitarte, D., Azpeitia, I., Barrera, O., Barrutieta, A., Blanco, H., and Echarte, F., Self-inspection based reproduction in cellular automata, in *Advances in Artificial Life* (eds. F. Moran et al.), LNAI-929, Springer-Verlag, 564–576 (1995).

[16] Laing, R., Automaton models of reproduction by self-inspection, *J. Theor. Biol.*, **66**, 437–456 (1977).