



**PROCEEDINGS OF  
THE FOURTH  
INTERNATIONAL SYMPOSIUM ON  
ARTIFICIAL LIFE AND ROBOTICS  
(AROB 4th '99)  
Vol. 2**

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

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

Proceedings of The Fourth International Symposium on  
**ARTIFICIAL LIFE AND ROBOTICS**  
(AROB 4th '99)

**Challenge for Complexity**

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

**Editors: Masanori Sugisaka and Hiroshi Tanaka**

**ISBN4-9900462-9-3, 1999**



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

**ORGANIZED BY**

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

**CO-SPONSORED BY**

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

**CO-OPERATED BY**

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

**SUPPORTED BY**

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

## HONORARY PRESIDENT

M. Hiramatsu (Governor, Oita Prefecture)

## HONORARY ADVISER

Y. Fujita (President, The Japan Academy)

## ADVISORY COMMITTEE CHAIRMAN

M. Ito (Director, RIKEN)

## ORGANIZING COMMITTEE

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



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

## STEERING COMMITTEE

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

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

## PROGRAM COMMITTEE

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



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

## LOCAL ARRANGEMENT COMMITTEE

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

## TOPICS

Hardware based topics are welcome  
in the fields given by

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

## COPYRIGHT

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



All correspondence related to the symposium  
should be addressed to :

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

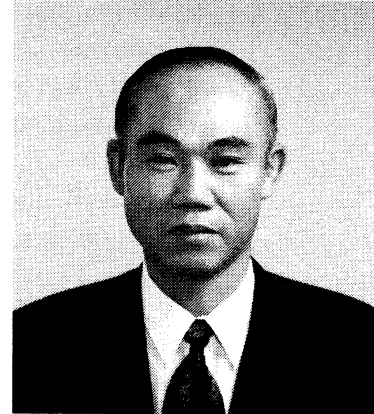
WWW Home Page <http://arob.cc.oita-u.ac.jp/>

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

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

## PREFACE

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



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

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

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

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

I am looking forward to meeting you in Beppu.

*Masanori Sugisaka*  
M. Sugisaka

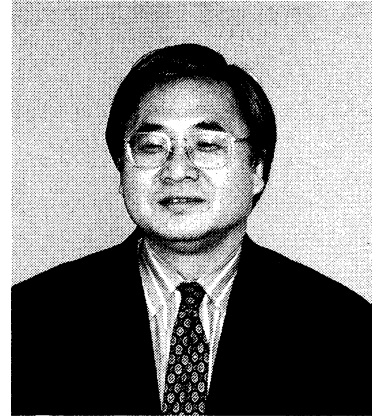
January 12, 1999



## PREFACE

### Hiroshi Tanaka

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



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

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

I am looking forward to meeting you in Beppu.

*Hiroshi Tanaka*  
H. Tanaka

January 12, 1999

# TECHNICAL PAPER INDEX

## Plenary and Open Symposium Lectures (Invited Talks)

- P1-1 *Computer simulation of gene flow in the malaria vector, anopheles gambiae, in West Africa* .....OP-1  
C. Taylor (University of California-Los Angeles, USA)
- P1-2 *Discussion of “living” machines using chaotic control* .....OP-2  
M. W. Tilden (Los Alamos National Laboratory, USA)
- P1-3 *Complexity and Control* .....OP-4  
H. Kimura (The University of Tokyo, Japan)
- OS-1 *Origins of life : from recent topics* .....OP-5  
T. Oshima (Tokyo University of Pharmacy and Life Science, Japan)
- OS-2 *The mind of the brain* .....OP-7  
G. Matsumoto (RIKEN, Japan)
- OS-3 *BizSim: The world of business in a abox* .....OP-15  
J. L. Casti (Santa Fe Institute, USA)
- P2 *A universal origin of life* .....OP-21  
S. Rasmussen, S. Colgate, K. Lachner, B. Lehnert, J. Solem, D. Whitten  
Los Alamos National Laboratory, USA)  
S. Rasmussen, S. Colgate (Santa Fe Institute, USA)  
L. Luisi (ETHZ Zurich, Switzerland)  
B. Mayer (University of Vienna, Austria)
- P3 *Evolving robot morphology and control* .....OP-22  
R. K. Belew, C. Mautner (University of California-San Diego, USA)
- P4 *Grasping impact force control of a flexible robotic gripper using piezoelectric actuator* .....OP-28  
W. R. Wells, W. Yim (University of Nevada-Las Vegas, USA)

P5 <i>Learning algorithms in a class of knowledge-based systems</i>	.....OP-34
Z. Bubnicki (Wroclow University of Technology, Poland)	
<b>Invited Lectures (Invited Talks)</b>	
IT1-1 <i>Evolution of differentiated multi-threaded digital organisms</i>	.....I-1
T. Ray, J. Hart (ATR, Japan)	
IT1-2 <i>Robot populations and their controlled evolution</i>	.....I-2
D. J. G. James (Coventry University, UK)	
V. G. Rumchev (Curtin University of Technology, Australia)	
IT1-3 <i>Ola: what goes up, must fall down</i>	.....I-9
H. H. Lund, J. A. Arendt, J. Fredslund, L. Pagliarini (University of Aarhus, Denmark)	
IT2-1 <i>Dynamic bidirectional associative memory using chaotic neurons</i>	.....I-16
J. J. Lee (KAIST, Korea)	
IT2-2 <i>A bottom-up way to develop multi-cellular digital organisms</i>	.....I-22
Y. G. Zhang, X. Wu (Academia Sinica, China)	
M. Sugisaka (Oita University, Japan)	
IT3-1 <i>Human-machine cooperative system for natural language processing</i>	.....I-27
R. Dai, Y. Fei (Chinese Academy of Science, China)	
IT3-2 <i>An adaptive classifier system tree for extending genetic based machine learning in dynamic environment</i>	.....I-31
D. Hu, R. Jiang, Y. Luo (Tsinghua University, China)	
IT4-1 <i>Winning strategies for robot war games</i>	.....I-35
B. Stilman (University of Colorado at Denver, USA)	
IT4-2 <i>Insect size robots with insect level intelligence</i>	.....I-41
J. D. Nicoud (Swiss Federal Institute of Technology, Switzerland)	

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

- A1-1 *Theater, movie with A-life -Romeo & Juliet in hades as A-life based cinema-* ..... 1  
N. Tosa (ATR, Japan)
- A1-2 *Treatment of nonverbal behaviors for A life-based computer actors* .....5  
R. Nakatsu, A. Solomides, N. Tosa (ATR, Japan)
- A1-3 *Technoetic theatre: performance and enactment in the dramaturgy of artificial life* .....12  
R. Ascott (University of Plymouth and Wales College, UK)
- A1-4 *Sociality of robots in symbiotic relations with humans* .....16  
T. Ono, M. Imai, T. Etani (ATR, Japan)

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

- B1-1 *Creation of subjective value through physical interaction between human and machine* .....20  
T. Shibata, K. Tanie (Ministry of International Trade and Industry, Japan)
- B1-2 *Task sharing in a multiple robot system controlled by macro-commands of an operator* .....24  
K. Ohkawa (University of Tsukuba, Japan)  
T. Shibata, K. Tanie (Ministry of International Trade and Industry, Japan)
- B1-3 *Evolutionary parallel collaborative computation on intelligent agents* .....28  
T. Yamaguchi, N. Kohata, M. Takahide, T. Baba (Utsunomiya University, Japan)  
H. Hashimoto (The University of Tokyo, Japan)
- B1-4 *Development of an intelligent data carrier (IDC) system and its applications* .....34  
D. Kurabayashi, H. Asama, T. Fujii, H. Kaetsu, I. Endo (RIKEN, Japan)
- B1-5 *Physical agent and media on computer network* .....40  
Y. Kunii (Chuo University, Japan)  
H. Hashimoto (The University of Tokyo, Japan)

B1-6 <i>Fast estimation of motion parameters for vehicle-camera using focus of expansion</i>	.....46
Z. Hu, K. Uchimura, S. Kawaji (Kumamoto University, Japan)	
B1-7 <i>Intelligent control of robots in coordination</i>	.....50
K. Kosuge, Y. Hirata, K. Takeo (Tohoku University, Japan)	
H. Asama, H. Kaetsu, K. Kawabata (RIKEN, Japan)	
<b>S-B2: Complexity (General Session) (Room B)</b>	
B2-1 <i>A study on coalition formation process in the iterated multiple lake game</i>	.....54
T. Yamashita, K. Suzuki, A. Ohuchi (Hokkaido University, Japan)	
B2-2 <i>Estimating an independent component based on the mutual information against the residual component</i>	.....58
T. Iwamoto (Mitsubishi Electric Corp., Japan)	
B2-3 <i>Rotational model of the stock prices(I)</i>	.....61
S. Maekawa, Y. Fujiwara (Communications Research Laboratory, Japan)	
B2-4 <i>Rotational model of the stock prices(II)</i>	.....65
Y. Fujiwara, S. Maekawa (Communications Research Laboratory, Japan)	
B2-5 <i>Multi agent approach against computer virus: An immunity-based system</i>	.....69
T. Okamoto (Nara Institute of Science and Technoloty, Japan)	
Y. Ishida (Toyohashi University of Technology, Japan)	
B2-6 <i>“Life Spacies”: a genetic text-to-form editor on the internet</i>	.....73
C. Sommerer, L. Mignonneau (ATR, Japan)	
<b>S-B3: Chaos engineering (Invited Session) (Room B)</b>	
B3-1 <i>Learning in software driven neural networks with temporal coding and functional connectivity</i>	.....78
M. Watanabe, K. Aihara (The University of Tokyo, Japan)	

B3-2 <i>Retrieval characteristics of associative chaotic neural networks with weighted pattern storage</i>	.....82
M. Adachi (Tokyo Denki University, Japan)	
K. Aihara (The University of Tokyo, Japan)	
B3-3 <i>Chaotic evolution in a game of a host and two parasites</i>	.....86
G. Hori (RIKEN, Japan)	
K. Aihara (The University of Tokyo, Japan)	
B3-4 <i>IC implementation of a multi-internal-state chaotic neuron model with unipolar and bipolar output functions</i>	.....90
Y. Horio, I. Kobayashi, H. Hayashi (Tokyo Denki University, Japan)	
K. Aihara (The University of Tokyo, Japan)	
B3-5 <i>Detecting nonlinear causality via nonlinear modeling</i>	.....94
T. Ikeguchi (Science University of Tokyo, Japan)	
B3-6 <i>Short-term prediction about complex sequences in a blast furnace by bell-shaped radial basis function networks</i>	.....98
T. Miyano, H. Shibuta (Sumitomo Metal Industries, Ltd., Japan)	
K. Aihara (The University of Tokyo, Japan)	
B3-7 <i>Some examples on reconstructing nonlinear dynamics by the generalized exponential autoregressive model</i>	.....102
Z. Shi, Y. Tamura, T. Ozaki (The Institute of Statistical Mathematics, Japan)	
<b>S-C1: Intelligent control system-discrete event systems approach</b>	
<b>(Invited Session) (Room C)</b>	
C1-1 <i>Autonomous distributed control by multi agent nets</i>	.....106
H. Nakajima, T. Miyamoto, S. Kumagai (Osaka University, Japan)	
C1-2 <i>Estimate based limited lookahead control of discrete event systems with model uncertainty</i>	.....110
S. Takai (Wakayama University, Japan)	



C1-3 <i>Conflict resolution in continuous petri nets using linear programming</i>	.....114
A. Tanaka, T. Ushio (Osaka University, Japan)	
S. Kodama (Kinki University, Japan)	
C1-4 <i>Performance improvement in an internal model control of discrete-event systems based on max-algebra</i>	.....118
S. Masuda, A. Inoue, Y. Hirashima (Okayama University, Japan)	
H. Suzuki (Toshiba Co., Ltd., Japan)	
C1-5 <i>Autonomous distributed control of manufacturing system by scene transition nets</i>	.....122
S. Kawata (Tokyo Metropolitan University, Japan)	
M. Haruyama (Mitsubishi Heavy Industries, Ltd., Japan)	
C1-6 <i>Introduction of biochemical allosteric property for creating a new electrical signal transmission system under the time minimum optimization</i>	.....126
H. Hirayama (Asahikawa Medical College, Japan)	
Y. Okita (National Institute of Special Education, Japan)	
<b>S-C2: Neural network and associative memory (Invited Session)</b>	
<b>(Room C)</b>	
C2-1 <i>Diffusively coupled multi-neural networks for optimization problems</i>	.....130
R. Horie, E. Aiyoshi (Keio University, Japan)	
C2-2 <i>Learning algorithm for intermodular connection of multimodular associative networks</i>	.....134
M. Mochizuki, H. Minamitani (Keio University, Japan)	
C2-3 <i>Multi-winners self-organizing multidirectional associative memory</i>	.....138
J. Huang, M. Hagiwara (Keio University, Japan)	
C2-4 <i>Bill money recognition by using the LVQ method</i>	.....142
T. Kosaka (Glory Inc., Japan)	
S. Omatsu (Osaka Prefecture University, Japan)	
C2-5 <i>Neuro-approach for hard disk driver position control</i>	.....146

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

C2-6 <i>A saccadic model with distributed feedback mechanism: simulations of normal and interrupted saccades</i>	.....150
--	----------

K. Arai (Mitsubishi Chemical Corp., Japan)

S. Das (ITT Systems & Sciences Corp., Japan)

E. L. Keller (University of California-Berkeley, USA)

E. Aiyoshi (Keio University, Japan)

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

C3-1 <i>Reinforcement learning approach to cooperative carrying problem</i>	.....154
---	----------

K. Kawakami, K. Ohkura, K. Ueda (Kobe University, Japan)

C3-2 <i>On the emergence of motion patterns in locomotion systems</i>	.....158
---	----------

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

C3-3 <i>Towards man-machine creativity in conceptual design</i>	.....162
---	----------

V. V. Kryssanov, H. Tamaki, S. Kitamura (Kobe University, Japan)

C3-4 <i>Self-organization process in interactive manufacturing environment</i>	.....166
--	----------

N. Fujii, I. Hatono, K. Ueda (Kobe University, Japan)

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

D1-1 <i>A intelligent control of robot manipulator by visual feedback</i>	.....170
---	----------

S. H. Han (Kyungnam University, Korea)

J. I. Bae (Pukyung National University, Korea)

M. H. Lee (Pusan National University, Korea)

D1-2 <i>Robust predictive control of robot manipulators</i>	.....175
---	----------

M. C. Han, M. H. Lee (Pusan National University, Korea)

D1-3 <i>Implementation of virtual reality through the fusion of visual and force information</i>	.....179
--	----------

S. K. An, S. J. Han, J. M. Lee, M. H. Lee (Pusan National University, Korea)

D1-4 <i>Hardware Implementation of fuzzy logic controller for speed of a DC series motor using an adaptive evolutionary computation</i>	.....183
---	----------

G. H. Hwang, K. J. Mun, J. H. Park, M. H. Lee (Pusan National University, Korea)

D1-5 <i>Development of a 3D graphic simulation tool for a SCARA robot</i>	.....188
---	----------

D. Y. Lee, J. W. Choi, M. H. Lee, K. Son, M. C. Lee, J. M. Lee (Pusan National University, Korea)

S. H. Han (Kyungnam University, Korea)

D1-6 <i>A polynomial fuzzy neural network for modeling and control</i>	.....192
--	----------

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

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

D2-1 <i>Development of horseback riding therapy simulator with VR technology</i>	.....196
--	----------

O. Sekine, Y. Shinomiya, R. Nakajima (Matsushita Electric Works, Ltd., Japan)

T. Kimura (Nippon Medical School, Japan)

D2-2 <i>A novel application of face image processing to wearable computer –augmentation of human memory for faces and names</i>	.....200
---	----------

H. Mizoguchi, T. Shigehara, Y. Goto, T. Mishima (Saitama University, Japan)

D2-3 <i>Virtual earphone to whisper in a person's ear remotely by utilizing visual tracking and speakers array</i>	.....204
--	----------

H. Mizoguchi, T. Shigehara, Y. Goto, M. Teshiba, T. Mishima (Saitama University, Japan)

D2-4 <i>Virtual concierge: a talking door-phone to sepak up visitor's name</i>	.....208
--	----------

H. Mizoguchi, T. Shigehara, Y. Goto, T. Mishima (Saitama University, Japan)

D2-5 <i>Realizing virtual wireless microphone to pick up human voice remotely and clearly</i>	.....212
---	----------

H. Mizoguchi, T. Shigehara, M. Teshiba, T. Mishima (Saitama University, Japan)

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

D3-1 <i>A neural network based artificial life model for navigation of multiple autonomous mobile robots in the dynamic environment</i>	.....216
S. K. Min, H. Kang (Chung-Ang University, Korea)	
D3-2 <i>Cellular automata based neural networks (CABANN) for optimal path planning</i>	.....220
Y. G. Jo, H. Kang (Chung-Ang University, Korea)	
D3-3 <i>Evolving cellular automata neural systems 2</i>	.....224
D. W. Lee, K. B. Sim (Chung-Ang University, Korea)	
D3-4 <i>Generalized asymmetrical BAM</i>	.....228
T. D. Eom, J. J. Lee (KAIST, Korea)	
D3-5 <i>Artificial immune system for realization of cooperative strategies and group behavior in collective autonomous mobile robots</i>	.....232
D. W. Lee, H. B. Jun, K. B. Sim (Chung-Ang University, Korea)	
D3-6 <i>Genetic programming-based A life techniques for evolving collective robotic intelligence</i>	.....236
D. Y. Cho, B. T. Zhang (Seoul National University, Korea)	
D3-7 <i>A digital artificial brain architecture for mobil autonomous robots</i>	.....240
A-P Uribe, E. Sanchez (Swiss Federal Institute of Technology-Lausanne, Switzerland)	
<b>S-E1: Complexity in life system (Invited Session) (Room E)</b>	
E1-1 <i>Complexities in biosystem</i>	.....244
H. Tanaka (Tokyo Medical and Dental University, Japan)	
E1-2 <i>The synthesis of gene with block automaton</i>	.....246
M. Kinoshita, M. Wada (Hokkaido University, Japan)	
E1-3 <i>Statistical approach to genetic algorithm</i>	.....250
Y. Fujiwara (Communications Research Laboratory, Japan)	

E1-4 <i>Toward the realization of an evolving ecosystem on cellular automata</i>	.....254
H. Sayama (The University of Tokyo, Japan)	
E1-5 <i>Prediction of deviant genetic codes -Why they evolve-</i>	.....258
T. Maeshiro (ATR, Japan)	
E1-6 <i>On a correlation between the degree of halting property and the qualitative behavior of abstract chemical system</i>	.....262
Y. Suzuki, H. Tanaka (Tokyo Medical and Dental University, Japan)	
<b>S-E2: Robotics I (General Session) (Room E)</b>	
E2-1 <i>Control of a humanoid robot using a multi-freedom motion capture device</i>	.....266
S. Kurono, Y. Miyamoto (Kyushu Sangyo University, Japan)	
S. Aramaki (Fukuoka University, Japan)	
E2-2 <i>Robot path search including obstacles by GA</i>	.....270
H. Yamamoto (Wakayama University, Japan)	
E2-3 <i>A robot control/learning scheme with task compatibility</i>	.....274
Q. Guo (Beijing Institute of Technology, China)	
E2-4 <i>Fine motion strategy using skill-based backprojection in consideration of uncertainty in control and sensing</i>	.....275
A. Nakamura, T. Suehiro, H. Tsukune (Electrotechnical Laboratory, Japan)	
T. Ogasawara (Nara Institute of Science and Technology, Japan)	
E2-5 <i>Sliding mode controller for robot manipulators with predetermined transient response</i>	.....279
K. B. Park, T. Tsuji (Kyushu Institute of Technology, Japan)	
J. J. Lee (KAIST, Korea)	
E2-6 <i>A novel application of legged mobile robot to human robot collaboration</i>	.....283
H. Mizoguchi, Y. Goto, K. Hidai, T. Shigehara, T. Mishima (Saitama University, Japan)	

E2-7 <i>Internal state acquisition for reinforcement learning agent by using radial basis function neural network</i>	.....287
H. Murao, S. Kitamura (Kobe University, Japan)	
<b>S-E3: Robotics II (General Session) (Room E)</b>	
E3-1 <i>Real-time search for autonomous mobile robot using the framework of anytime algorithm</i>	.....291
K. Fujisawa, T. Suzuki, S. Okuma (Nagoya University, Japan)	
S. Hayakawa (Toyota Technological Institute, Japan)	
T. Aoki (Nagoya Municipal Industry Research Institute, Japan)	
E3-2 <i>Incremental evolution of CAM-brain to control a mobile robot</i>	.....297
G. B. Song, S. B. Cho (Yonsei University, Korea)	
E3-3 <i>Path planning for mobile robot using a genetic algorithm</i>	.....301
S. Tamura, M. Takuno, T. Hatanaka, K. Uosaki (Tottori University, Japan)	
E3-4 <i>An evolvable NAND-logic circuit applied to a real mobile robot khepera</i>	.....305
Y. Wei, M. M. Islam, R. Odagiri, T. Asai, K. Murase (Fukui University, Japan)	
E3-5 <i>Cooperation of real mobile robots using communication</i>	.....309
M. M. Islam, Y. Wei, R. Odagiri, T. Asai, K. Murase (Fukui University, Japan)	
E3-6 <i>Solving the equations of constrained motion in a lower LIMB model</i>	.....313
C. Itiki (University of San Paulo, Brazil)	
R. Kalaba (University of Southern California, USA)	
H. Natsuyama (Seasons Associates, USA)	
E3-7 <i>Estimation of muscle parameters of a lower LIMB model</i>	.....317
C. Itiki (University of San Paulo, Brazil)	
R. Kalaba (University of Southern California, USA)	
H. Natsuyama (Seasons Associates, USA)	
<b>S-A2: Emergent system design II (Invited Session) (Room A)</b>	



A2-1 <i>A principle of design of an autonomous mobile robot</i>	.....320
K. Tsuchiya, K. Tsujita (Kyoto University, Japan)	
A2-2 <i>Autonomous robot control by a neural network with dynamic rearrangement function</i>	.....324
T. Kondo, A. Ishiguro, Y. Uchikawa (Nagoya University, Japan)	
P. Eggenberger (University of Zurich, Switzerland)	
A2-3 <i>Maintenance of diversity by means of thermodynamical selection rules for genetic problem solving</i>	.....330
H. Kita (Tokyo Institute of Technology, Japan)	
N. Mori (Osaka Prefecture University, Japan)	
Y. Nishikawa (Osaka Institute of Technology, Japan)	
A2-4 <i>Protein folding by a hierarchical genetic algorithm</i>	.....334
O. Takahashi, H. Kita, S. Kobayashi (Tokyo Institute of Technology, Japan)	
<b>S-B4: Artificial life (General Session) (Room B)</b>	
B4-1 <i>Artificial behavior of cell-like structure with polarized elements</i>	.....340
T. Kohashi, T. Takayanagi, K. Suzuki, A. Ohuchi (Hokkaido University, Japan)	
B4-2 <i>Origin and evolution of early peptide-synthesizing biomachines by means of hierarchical sociogenesis of intracellular primitive tRNA-riboorganisms</i>	.....344
K. Ohnishi, S. Hokari (Niigata University, Japan)	
H. Yanagawa (Mitsubishi Kasei Institute of Life Sciences, Japan)	
B4-3 <i>Long-term increase of complexity and functional diversification by contingent mutations in computational algorithms</i>	.....350
S. Ohashi, Y. Kakazu (Hokkaido University, Japan)	
S. Yoshii (University of Liverpool, UK)	
B4-4 <i>Team plays of soccer agents based on evolutionary dynamic formations</i>	.....354
T. Murata, M. Yamamoto, K. Suzuki, A. Ohuchi (Hokkaido University, Japan)	

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

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

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

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

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

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

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

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

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

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

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

B6-1 <i>Immune algorithm with immune network and major histocompatibility complex</i>	.....391
N. Toma, S. Endo, K. Yamada (University of the Ryukyus, Japan)	
B6-2 <i>Application of competitive co-evolution algorithm to iterated prisoner's dilemma</i>	.....395
M. Nerome, S. Endo, K. Yamada, H. Miyagi (University of the Ryukyus, Japan)	
B6-3 <i>Vector lyapunov functions method in stability and control theories for logic-dynamical systems</i>	.....399
V. M. Matrosov (Russian Academy of Science, Russia)	
B6-4 <i>AHP coefficients optimization technique based on GA</i>	.....403
T. Toma, M. R. Asharif (University of the Ryukyus, Japan)	
B6-5 <i>Study on cooperative agents through adaptive focal point</i>	.....407
S. Yamauchi, K. Yamada, S. Endo, H. Miyagi (University of the Ryukyus, Japan)	
B6-6 <i>On schemes for analysis of stability and asymptotical estimations in critical cases of stability theory</i>	.....411
I. V. Matrosov (Moscow State University, Russia)	
<b>S-C4: Genetic algorithms I (General Session) (Room C)</b>	
C4-1 <i>Parallel distributed architectures of biologically inspired parameter-free genetic algorithm for simulating ecosystems</i>	.....415
H. Sawai, S. Adachi (Kansai Advanced Research Center, Japan)	
S. Kizu (Toshiba R & D Center, Japan)	
C4-2 <i>A distributed system inspired from the immune system: an application to control</i>	.....419
Y. Ishida (Toyohashi University of Technology, Japan)	
C4-3 <i>Evolution of vision system for stereo perception by genetic algorithm</i>	.....423
W. Nian, K. Okazaki (Fukui University, Japan)	
S. Tamura (Osaka University, Japan)	
C4-4 <i>Immune algorithm with adaptive memory</i>	.....427
M. Yonezu, T. Yoshida, M. Nakanishi (Keio University, Japan)	

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

- C5-1 *Neurocontroller for load swing suppression of a jib crane on a floating bed* .....431  
F. Tabuchi, E. Uezato, H. Kinjo, T. Yamamoto (University of the Ryukyus, Japan)
- C5-2 *Self-organization of emotional neural network* .....435  
H. Kinjo, H. Ochi, T. Yamamoto (University of the Ryukyus, Japan)
- C5-3 *Intelligent maneuvering of a flight-type wall-climbing robot* .....439  
H. Miyagi, A. Nishi (Miyazaki University, Japan)
- C5-4 *Block fuzzy neural networks for controlling an industrial manipulator* .....440  
J. Tang, K. Kuribayashi (Yamaguchi University, Japan)
- C5-5 *A fuzzy model to control the temperature in cooling of metal molds with spray robot* .....444  
T. Sakamoto, K. Murakami (Ube Industries Co. Ltd., Japan)  
K. Kuribayashi (Yamaguchi University, Japan)

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

- C6-1 *Improved genetic algorithm for generalized transportation problem* .....448  
M. Gen, K. Ida, Y. Z. Li, J. Choi (Ashikaga Institute of Technology, Japan)
- C6-2 *Genetic algorithms approach on leaf-constrained spanning tree problem* .....452  
G. Zhou, M. Gen (Ashikaga Institute of Technology, Japan)
- C6-3 *A genetic algorithm for bicriteria fixed charge transportation problem* .....456  
Y. Z. Li, M. Gen, K. Ida (Ashikaga Institute of Technology, Japan)
- C6-4 *A spanning tree-based genetic algorithm for reliable multiplexed network topology design* .....460  
J. R. Kim, M. Gen, K. Ida (Ashikaga Institute of Technology, Japan)

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

- D4-1 *Toward emergent intelligence in multiagent learning* .....464  
K. Takadama, K. Shimohara (ATR, Japan)  
T. Terano, (University of Tsukuba, Japan)  
K. Hori, S. Nakasuka (The University of Tokyo, Japan)
- D4-2 *Behaviors of pedestrians with learning ability in various underground areas* .....468  
M. H. Zheng, Y. Kashimori, T. Kambara (The University of Electro-Communications, Japan)

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

- D5-1 *A framework for extending classifier systems under dynamic learning environments* .....472  
R. Jiang, Y. Luo, D. Hu, H. Xi (Tsinghua University, China)
- D5-2 *Quantum tunneling evolution: A model of life as a global optimization process* .....476  
M. Hirafuji, S. Hagan (National Agriculture Research Center, Japan)
- D5-3 *Automatic parallelization of sequential programs using genetic programming* .....480  
C. Ryan, L. Ivan (University of Limerick, Ireland)

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

- D6-1 *A global search method for all roots of algebraic equations by genetic algorithm* .....484  
S. Yamada, I. Yoshihara, K. Ozawa, K. Abe (Tohoku University, Japan)
- D6-2 *Optimal control method using genetic algorithm and its application* .....488  
T. Nishimura, K. Sugawara, I. Yoshihara, K. Abe (Tohoku University, Japan)
- D6-3 *Optimization of foraging behavior by interacting multi-robots* .....492  
K. Sugawara, I. Yoshihara, K. Abe (Tohoku University, Japan)
- D6-4 *Optimization of delivery route in a city area using genetic algorithm* .....496  
A. Takeda, S. Yamada, K. Sugawara, I. Yoshihara, K. Abe

B. (Tohoku University, Japan)

D6-5 *Time series prediction modeling by genetic programming without inheritance of model parameters* .....500

M. Numata, K. Sugawara, S. Yamada, I. Yoshihara, K. Abe (Tohoku University, Japan)

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

E4-1 *Information transformation by virus-evolutionary genetic programming* .....504

N. Kubota, F. Kojima, S. Hashimoto (Osaka Institute Technology, Japan)  
T. Fukuda (Nagoya University, Japan)

E4-2 *A description of dynamic behavior of sensory/motor systems with fuzzy symbolic dynamic systems* .....508

I. Takeuchi, T. Furuhashi (Nagoya University, Japan)

E4-3 *Intelligent fault tolerant system of vibration control for flexible structures* .....512

M. Isogai (Yamazaki Mazak Co., Japan)  
F. Arai, T. Fukuda (Nagoya University, Japan)

E4-4 *An evolutionary technique for constrained optimization problems* .....516

M. M. A. Hashem, K. Watanabe, K. Izumi (Saga University, Japan)

E4-5 *Evolving in dynamic environments through adaptive chaotic mutation* .....520

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

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

E5-1 *Organizational evolution by learning adaptive functions* .....524

M. Ishinishi, A. Namatame (National Defense Academy, Japan)

E5-2 *An Evolutionary design of commitment networks* .....528

K. Uno, A. Namatame (National Defense Academy, Japan)

E5-3 <i>The evolutionary approach for designing a virtual organization</i>	.....532
Y. Shimoyama, A. Namatame (National Defense Academy, Japan)	
E5-4 <i>What facilitates emergence of symbiosis</i>	.....536
M. Chang, K. Ohkura, K. Ueda (Kobe University, Japan)	
E5-5 <i>Synthetic collective behavior by multiple reinforcement learning agents in simulated dodgeball game</i>	.....540
N. Ono, S. Yoshida (University of Tokushima, Japan)	
E5-6 <i>Evolutionary design of analog electronic circuits</i>	.....544
H. Shibata, S. Samadi, H. Iwakura (The University of Electro-Communications, Japan)	
<b>S-E6: Chaos (General Session) (Room E)</b>	
E6-1 <i>Periodic motion generated after chaos in a model of trading agents</i>	.....548
M. Tanaka-Yamawaki, M. Tabuse (Miyazaki University, Japan)	
E6-2 <i>A study of chaos associative memory</i>	.....552
M. Nakagawa (Nagaoka University of Technology, Japan)	
E6-3 <i>A chaotic synthesis model of vowels</i>	.....556
H. Koga, M. Nakagawa (Nagaoka University of Technology, Japan)	
E6-4 <i>A chaos model to solve the optimal stable marriage problem</i>	.....560
R. Hiroi, M. Nakagawa (Nagaoka University of Technology, Japan)	
<b>S-A3: Perception in vision and hearing (Invited Session) (Room A)</b>	
A3-1 <i>Computation of motion direction by output neurons of the retina</i>	.....564
H. Uchiyama (Kagoshima University, Japan)	
A3-2 <i>Hysteresis phenomena in depth perception of a moving object</i>	.....568
H. Jinnai, T. Kitazoe, T. Shii (Miyazaki University, Japan)	



A3-3 <i>Visual perception depends on auditory stimuli?</i>	.....572
K. Manabe, H. Riquimaroux (Doshisha University, Japan)	
A3-4 <i>Speech recognition using stereovision neural network model</i>	.....576
T. Kitazoe, S. I. Kim, T. Ichiki (Miyazaki University, Japan)	
A3-5 <i>Pitch perception in the Japanese macaque: Physiological and behavioral approach</i>	.....580
H. Riquimaroux, K. Manabe (Doshisha University, Japan)	
A3-6 <i>Hands-free speech recognition in echo and noise environments</i>	.....584
S. I. Kim, T. Kitazoe (Miyazaki University, Japan)	
<b>S-A4: Artificial brains etc. (General Session) (Room A)</b>	
A4-1 <i>Artificial mind in a classical context</i>	.....588
S. Hagan, M. Hirafuji (N. A. R. C., Japan)	
A4-2 <i>A computational model of viewpoint-forming process in searching solutions</i>	.....592
<i>- theory and methods using hierarchical classifier system-</i>	
T. Yoshimi, T. Taura (The University of Tokyo, Japan)	
A4-3 <i>An evolutionary architecture for a humanoid robot</i>	.....598
P. Nordin, M. G. Nordahl (Chalmers University of Technology, Sweden)	
A4-4 <i>A development of computer aided identification for systems under control</i>	.....602
K. Oura, T. Murakoshi, K. Akizuki (Waseda University, Japan)	
I. Hanazaki (Tokyo Denki University, Japan)	
A4-5 <i>ATR's artificial brain (CAM-Brain ) project: A sample of what individual CoDi-1Bit model evolved neural net modules can do</i>	.....606
H. D. Garis, N. E. Nawa (ATR, Japan)	
M. Korkin (Genobyte Inc., USA)	
F. Gers (Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Switzerland)	
M. Hough (Stanford University, USA)	

A4-6 <i>Spiker: Analog waveform to digital spiketrain conversion in ATR's artificial brain (CAM-Brain) project</i>	.....610
--	----------

M. Hough (Stanford University, USA)

H. D. Garis, N. E. Nawa (ATR, Japan)

M. Korkin (Genobyte Inc., USA)

F. Gers (Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Switzerland)

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

B7-1 <i>Generation of jumping motion pattern for hopping robot using genetic algorithm</i>	.....614
--	----------

Y. Yoshida, T. Kamano, T. Yasuno, T. Suzuki (The University of Tokushima, Japan)

Y. Kataoka (Kataoka Machine Co., Ltd., Japan)

B7-2 <i>An evolutionary optimal obstacle avoidance method for mobile robots</i>	.....618
---	----------

M. M. A. Hashem, K. Watanabe, K. Izumi (Saga University, Japan)

B7-3 <i>Design and experiment of an omnidirectional mobile robot using fuzzy servo control</i>	.....622
--	----------

J. Tang, K. Sanefuji (Yamaguchi University, Japan)

K. Watanabe (Saga University, Japan)

B7-4 <i>An experiment on force control using fuzzy environment models</i>	.....626
---	----------

F. Nagata (Fukuoka Industrial Technology Center, Japan)

K. Watanabe, K. Izumi, K. Sato, S. Akama (Saga University, Japan)

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

B8-1 <i>Realization of artificial human decision making based on conditional probability</i>	.....630
--	----------

M. Nakamura, S. Goto, T. Sugi (Saga University, Japan)

B8-2 <i>Artificial life with play instinct</i>	.....636
--	----------

S. Tamura, (Osaka University, Japan)

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

B8-3 *Autocatalysis as internal measurement and origin of programs* .....640

S. Toyoda (Kobe University, Japan)

B8-4 *An immuno system model for generation of self tolerance and memory* .....644

Y. Ochi, Y. Kashimori, T. Kambara (The University of Electro-Communications, Japan)

B8-5 *Simulations of group action of artificial honey bees* .....648

Y. Niino (Johokagaku High School, Japan)

M. Sugisaka (Oita University, Japan)

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

C7-1 *Robust and fast color-detecting using a look-up table* .....650

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

C7-2 *The multi-agent system's design based on behavior-based learning model* .....654

X. Wang, M. Sugisaka (Oita University, Japan)

C7-3 *An intelligent control strategy for robot soccer* .....658

T. Y. Kuc, S. M. Baek, I. J. Lee, K. O. Sohn (Sung Kyun Kwan University, Korea)

C7-4 *The design and development of robot soccer player* .....662

C. Zhang, C. Qu, D. Z. Gao (Northeastern University, China)

C7-5 *Simulation model of micro-robot soccer system* .....666

F. Rui, X. Liang, X. Xu (Northeastern University, China)

C7-6 *Role level design in a hybrid control structure for a vision-based soccer robot system* .....670

H. S. Shim, M. J. Jung, H. S. Kim, J. H. Kim, P. Vadakkepat (KAIST, Korea)

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

C8-1 <i>Bio-Informatic coordination and interaction among artifacts and humans</i>	.....674
O. Katai, T. Sawaragi (Kyoto University, Japan)	
C8-2 <i>A basic study of virtual collaborator – the first prototype system integration</i>	.....682
H. Ishii, W. Wu, D. Li, H. Ando, H. Shimoda, H. Yoshikawa (Kyoto University, Japan)	
T. Nakagawa (Mitsubishi Electric Corp., Japan)	
C8-3 <i>Generating novel memories by integration of chaotic neural network modules</i>	.....686
A. Sano (Kyoto University, Japan)	
 <b>S-D7: Neural networks I (General Session) (Room D)</b>	
D7-1 <i>Constrained hierarchical path planning of a robot by employing neural nets</i>	.....690
S. Patnaik, A. Konar, A. K. Mandal (Jadavpur University, India)	
D7-2 <i>A new learning method using prior information of neural networks</i>	.....694
B. Lu, K. Hirasawa, J. Murata, J. Hu (Kyushu University, Japan)	
D7-3 <i>Design of neural network controller using feedback structure</i>	.....699
W. J. Shin, S. Y. Lee (Kyungnam University, Korea)	
K. Hirasawa (Kyusyu University, Japan)	
D7-4 <i>Electrical equivalent circuit and resting membrane potential of neuron</i>	.....703
X. Zhang, H. Wakamatsu (Tokyo Medical and Dental University, Japan)	
D7-5 <i>An Adaptive associative memory system based on autonomous reaction between image memories</i>	.....707
Y. Kinouchi, M. Mizutani, A. Satou, F. Shouji (Tokyo University of Information Science, Japan)	
S. Inabayashi (System Sogo Kaihatsu Co., Ltd., Japan)	
D7-6 <i>Human-face recognition using neural network with mosaic pattern</i>	.....711
H. Kondo, S. B. A. Rahman (Kyushu Institute of Technology, Japan)	

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

D8-1 <i>Diagnosis with fuzzy belief networks</i>	.....715
I. Chakraborty (Mie University, Japan)	
A. Konar, A. K. Mandal (Jadavpur University, India)	
D8-2 <i>Design of output feedback controllers for Takagi-Sugeno fuzzy descriptor systems</i>	.....719
J. Yoneyama, K. Itoh, A. Ichikawa (Shizuoka University, Japan)	
D8-3 <i>Stability study of fuzzy logic control system for an inverted pendulum</i>	.....723
K. Nakano (Fukuoka Institute of Technology, Japan)	
M. Tomizuka (University of California at Berkeley, USA)	
D8-4 <i>Application of neural and fuzzy control strategies for a mobile vehicle</i>	.....729
X. Wang, M. Sugisaka (Oita University, Japan)	
<b>S-E7: Neural networks II (General Session) (Room E)</b>	
E7-1 <i>The time minimum optimization strategy of idiotypic immune type signal transmission network system</i>	.....733
H. Hirayama (Asahikawa Medical College, Japan)	
Y. Okita (National Institute of Special Education, Japan)	
E7-2 <i>An application of multi-modal neural network to multiple control system</i>	.....737
T. Nakagawa, K. Sugawara, I. Yoshihara, K. Abe (Tohoku University, Japan)	
M. Inaba (Hitachi Ltd., Japan)	
E7-3 <i>Dynamical recognition via hybrid neural networks</i>	.....741
N. Honma, K. Abe (Tohoku University, Japan)	
H. Takeda (Tohoku Gakuin University, Japan)	
E7-4 <i>Incremental evolution of neural controllers for navigation in a 6-legged robot</i>	.....745
D. Filliat, J. Kodjabachian, J-A. Meyer (AnimatLab, France)	
E7-5 <i>Pulse neural network applied to the binding problem-Implementation of decision making of the mobile robot-</i>	.....751

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

E7-6 *Research on using dynamic neural networks in model predictive control* .....755

S. R. Li, F. Li (University of Petroleum, China)  
Q. Lu (Tsinghua University, China)

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

E8-1 *Adaptive positioning of soccer agents with hybrid learning system* .....759

N. Akiyama, K. Suzuki, M. Yamamoto, A. Ohuchi (Hokkaido University,  
Japan)

E8-2 *Recent design of motion planner and behaviors for soccer-playing robots* .....763

S. G. Hong, T. D. Eom, C. Y. Lee, M. S. Kim, J. J. Lee (KAIST, Korea)

E8-3 *Neural network architecture optimization and application using genetic algorithm* .....767

Z. J. Liu, M. Sugisaka (Oita University, Japan)



# A Shape Input System for Three Dimensional Object in the Virtual Space

Jing-Long Wu, Masayuki Kitazawa and Hidekazu Harada

Department of Mechanical Engineering, Yamaguchi University  
Tokiwadai 2557,Ube 755-8611, Japan  
Email:wu@mechgw.mech.yamaguchi-u.ac.jp

## Abstract

Recently, on acquiring a shape of an object, various methods have been developed. In these methods, a solid model that is a target is necessary. A shape input device which does not need the solid model is our goal. As a first step, we propose a shape input system using the solid model. For this system, the spatial position sensor is redesigned to be compact. Therefore, the proposed shape input system has a simple structure and low cost. In this paper, we examined its performance through some basic experiments and acquired a target's shape. These results are sufficient accurate for shape input system.

**Key Word:** *Virtual reality, Spatial position sensor, Three dimensional shape input system, Virtual space*

## 1. Introduction

With the progress of the computer technology, CAD/CAM technology has been developed. But on acquiring a shape of an object, especially in the art and the design field, an object is actually made in advance and it is measured by an input device of the three dimensional(3-D) shape. Thus, a solid model is necessary in order to acquire its shape. Our idea is that making a target's shape by using hand's movements substitute for making a target's solid model. Up to now, several types of 3-D shape input device have been developed. They use joint arm<sup>1</sup>, data glove<sup>1</sup> and wire<sup>2</sup>. The type of using joint arm has small measurement area. The type of using data glove is very expensive. The type of using wire has simple structure, but it uses four wires to obtain spatial position. For this reason, this device is not suitable for complicated hand movements. From these reasons, we adopt the spatial position sensor<sup>3</sup> which uses only one wire to develop a 3-D shape input device.

As a first step, we propose a shape input system using the spatial position sensor for 3-D solid model. In this system, the spatial position sensor is redesigned to be a decreased size and increased measurement area. Therefore, this system has a large measurement area and low cost. The proposed system has another device. It is a rotary device on which a target(3-D solid model) is set. The target is rotated on this device. So, we can get the target's shape data in every point.

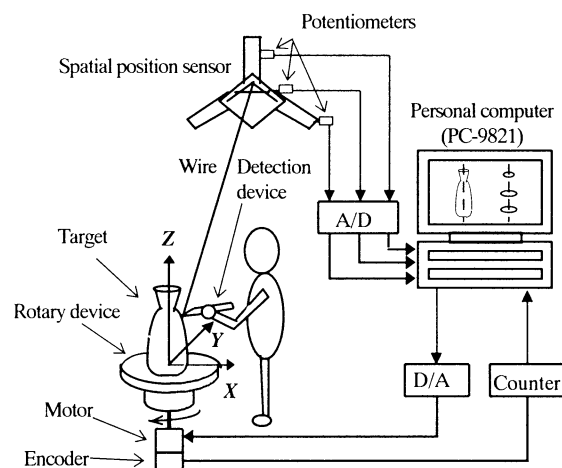
In this paper, the performance of the redesigned spatial position

sensor is demonstrated through some basic experiments and the proposed shape input system is examined on its performance by using a 3-D target.

## 2. The Proposed Shape Input System

As shown in Fig. 1, the shape input system is set in normal geodesic polar coordinates. A shape of a target is expressed in the coordinates. The progress for acquiring the shape of the target is explained in this section.

First, the target is set and rotated on the rotary device that is driven by a motor(SANYO R-406-011E). At this time, the rotary device's revolution is controlled by a personal computer(NEC PC9821) by way of a D/A board. Because of counting its revolution, any spatial position on the target's surface can be obtained. Next, he/she has the pen shaped detective device which is connected to the spatial position sensor by the wire and touches it to the target. This device is set at an initial position. He/She traces the detective device on the target's surface from bottom to top. Then the detective device's movement from the initial position is measured by the spatial position sensor. The output of this sensor is brought to the personal computer through an A/D board. In this computer, the target's shape at any section is shown on the CRT based on the revolution of the rotary device and the output of the spatial position sensor.



**Fig.1** Schematic of the shape input system for three dimensional object



### 3. Characteristic of The Redesigned Spatial Position Sensor

#### 3.1 The Redesigned Spatial Position Sensor

As described above, the size of the spatial position sensor was redesigned to be compacted. This spatial position sensor's size is reduced from  $770 \times 760 \times 200$  [mm] ( $W \times D \times H$ ) to  $660 \times 400 \times 200$  [mm] and the sensor's weight is also reduced to 4 [Kg]. Fig.2 shows the redesigned spatial position sensor.

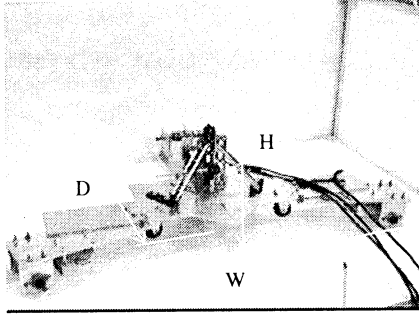


Fig.2 General view of the redesigned spatial sensor

#### 3.2 Feasible Measurement Area of $\alpha$ and $\beta$

The spatial position of the point  $P(r, \alpha, \beta)$  at the detection device is calculated from the length  $r$ , angles  $\alpha$  and  $\beta$ . The length  $r$  is directly measured by a potentiometer. The angles  $\alpha$  and  $\beta$  can be calculated from a displacement of the sliding bar which is kept with a movement of the detection device. This displacement is measured by other two potentiometers. The calculated algorithms are explained in reference 3.

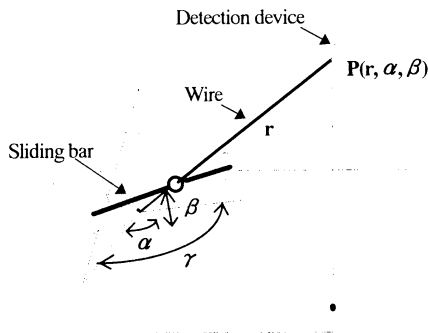


Fig. 3 The structure of the spatial position sensor

As shown in Fig.3, the angle  $\gamma$  mainly influences the feasible measurement area of the  $\alpha$ . Therefore, the angle  $\gamma$  is increased from  $90[\text{deg.}]$  to  $120[\text{deg.}]$  to obtain larger feasible measurement area than that of the original sensor. The feasible measurement area is examined experimentally. The measurement area of the  $\alpha$  and

$\beta$  are shown as follows:

$$\begin{aligned} 28.3(21.3) \leq \alpha \leq 79.6(64.2) & \quad [\text{deg.}] \cdots (1) \\ 6.0(23.5) \leq \beta \leq 51.7(58.2) & \quad [\text{deg.}] \cdots (2) \end{aligned}$$

Here, values within ( ) indicate the feasible measurement area of the original sensor. The feasible measurement area of this redesigned sensor is wider than that of the original one. To acquire static and dynamic characteristics, we examined some basic experiments. These results are shown below.

#### 3.3 Static Characteristic

The spatial position sensor fitting to the shape input system is shown in Fig.4. In this situation, the frames for setting the measurement points are equipped and 8 measurement points are set on them. These measurement points are arranged spirally and they are measured 5 times at each point. Fig.5 shows these results at  $X$ - $Z$ ,  $X$ - $Y$  and  $Z$ - $Y$  planes, respectively. Every point in Fig.5 shows the average of 5 times. In order to evaluate the static accuracy of the spatial position sensor, the differences between the actual values and the measured values are calculated from Fig.5. And They are shown on Table 1.

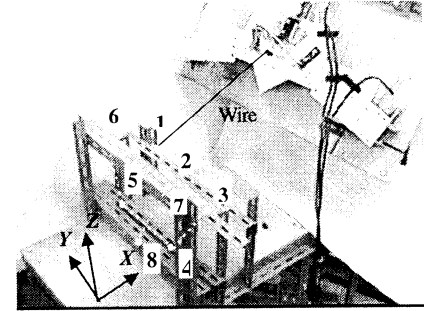


Fig. 4 Eight spiral measurement points in the shape input system

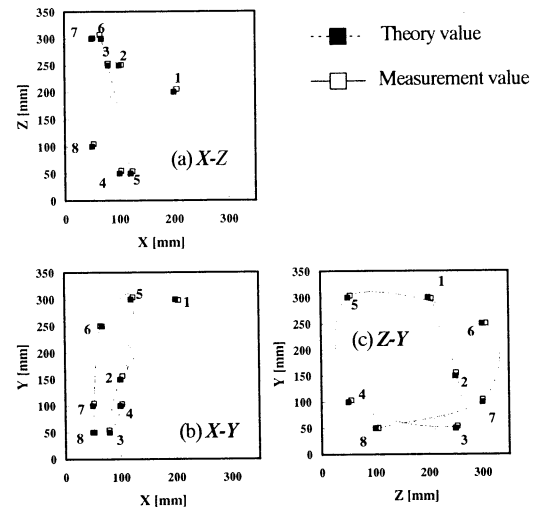


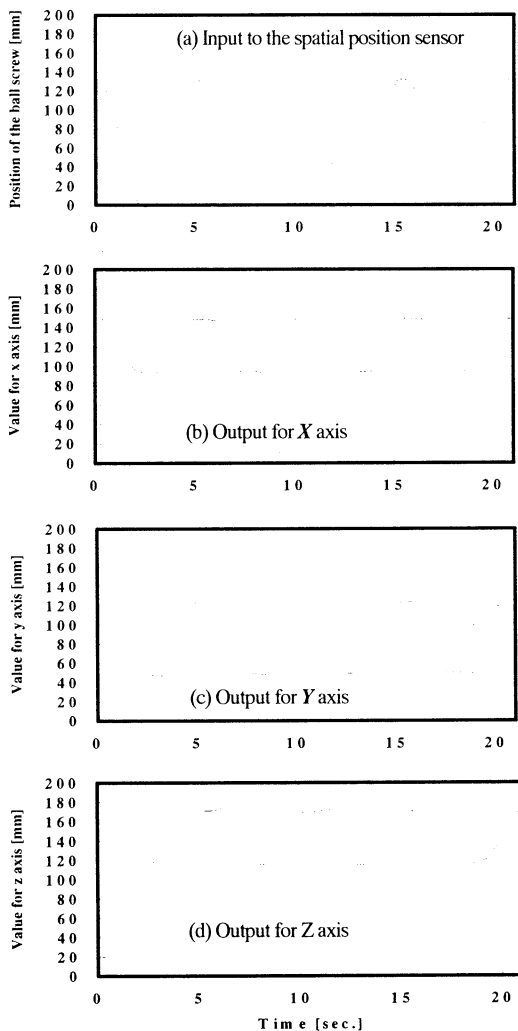
Fig.5 Measurement results for static characteristic

**Table 1 Static errors**

Measurement point	X axis(%)	Y axis(%)	Z axis(%)
1	2.7	-0.6	2.8
2	4.0	4.3	0.5
3	-0.1	8.4	1.4
4	3.2	3.5	10.4
5	3.0	1.3	9.0
6	-4.0	0.2	2.5
7	4.6	4.5	0.3
8	5.8	0.4	5.0
Average of error	2.4	2.8	4.0

### 3.4 Dynamic Characteristic

We examined the redesigned spatial position sensor's dynamic characteristic. A ball screw with a motor(R404T-0011,SANYO) is employed as a device that gives some time displacements to the spatial position sensor. This sensor's wire is fitted to the ball screw. Therefore, when the motor moves the ball screw, a displacement is given to the spatial position sensor. The given displacement and values measured by the spatial position sensor are shown in Fig. 6, respectively. The frequency of this displacement is 0.18[Hz].



**Fig.6 Measurement results for dynamic characteristic**

### 3.5 Results and Discussion

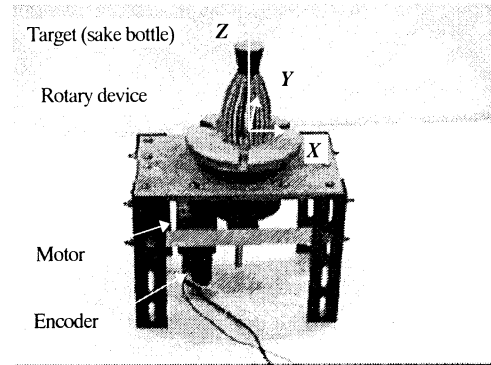
As shown in Fig.5 and Table 1, the errors of the sensor in X, Y and Z axes are within about 4%. As the first step, these errors are adequate for the shape input sensor.

As shown in Fig.6, there is some delay in every axis. They are 0.3, 0.3 and 0.2[sec.] in X, Y and Z axis, respectively. In the shape input device, a tracing time for inputting an object's shape is about 5[sec.]. So, these delay do not influence this system's response.

## 4. Shape Input Experiment

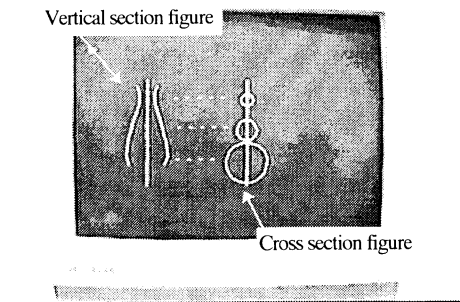
In this section, we describe about the shape input experiment by using the proposed shape input system. When a target on the rotary device is rotating, he/she traces it by using the detection device. At this time, the shape data of the target is obtained and these data are expressed on the personal computer's CRT. In results and discussion of this section, those shapes are compared with the real shape.

### 4.1 Methods



**Fig. 7 Target on the rotary device**

The target(sake bottle 125×60[mm]) and the rotary device are shown in Fig. 7. The target is rotated at 30[rpm]. The detection device traces the surface of the target from its bottom to top. It takes about 5[sec.]. At this time, the proposed shape input system obtain the target's spatial positions in the geodesic polar coordinates, measurement time and angles for the target's rotation as data. To do so, this system can make figures at any section. Once those data are stored at a hard disk equipped with the personal computer. After tracing the target, the measured shape is expressed as 2-D vertical and cross section figures on the CRT by using the acquiring data. These sections figures are composed by the measurement points and the straight lines connecting the measurement points as shown in Fig.8. Here, the proposed shape input system at this version can express only the 2-D section figure based on measured data. So, the next version of this system will have the performance to express 3-D figure.

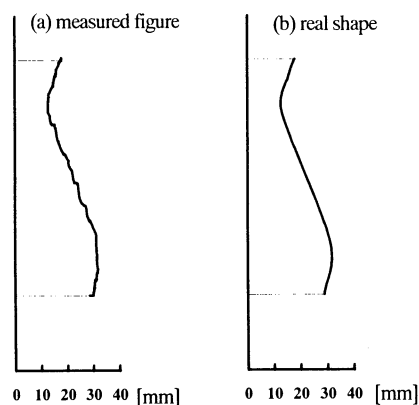


**Fig. 8 Vertical and cross section figures on CRT**

## 4.2 Results and Discussion

The real target's shape and the vertical section figure are obtained by the proposed shape input system are shown in Fig. 9. These two figures are very similar to each other. But, there are some waving parts in the measured vertical section figure. As a cause of these, the sliding bar does not follow the movement of the detection device smoothly. The errors between the real shape and the measured vertical section figure is 2.4%(about 1.5[mm]). This errors are similar to the static experiment's results. Also the errors are similar to the errors of the other type 3-D shape input system using wire<sup>2</sup>(about 2[mm]) and lights<sup>4</sup> (1.5% at cross section). Because of these reasons, this proposed shape input system has enough accuracy for getting a shape of 3-D object.

As described in 4.1 Methods, the measured shape is expressed as 2-D vertical and cross section figures on the CRT in this system. The method is not enough that the target is expressed as 3-D object. But the data which is acquired by the proposed system have enough information to make up 3-D object, because this system can obtain the vertical section data at any angle of the target's rotation and the cross section data at any value for Z axis as shown in Fig. 8. So, as this problem depends on software, we will program a method to express as 3-D object in the next version.



**Fig. 9 The measured figure and real shape at the vertical section**

## 5. Conclusion

In this paper, we proposed the shape input system using the spatial position sensor. For this system, the spatial position sensor was redesigned. We examined its performance through some basic experiments. Besides we measured the target's shape by using this shape input system, and its vertical and cross section figures were shown on the CRT. From these results, we obtained some conclusions as follows:

- 1) The redesigned spatial position sensor is more compact than the original one. This reduces low cost.
- 2) The redesigned spatial position sensor has sufficient accuracy for the static and dynamic characteristic.
- 2) The proposed shape input system for 3-D object has sufficient accuracy for inputting the solid model's shape.
- 3) The expression method is not real. There is a need to develop a new method that expresses 3-D figure.

Our future study is to improve this system to deal with complex object such as a cup with handle. Also, there is a need to express a measured shape in real time.

## References

1. Nissho Electronics Corporation (1996), Virtual Reality Products Guid. pp.7-12
2. Sato M, Hirota Y and Kawahara H (1991), Space Interface Device for Artificial Reality - SPIDAR- (in Japanese). Transaction of Institute of Electronics, Information and Communication Engineers Vol J74- D- II No.7,1991, pp.887-894
3. Kitazawa M, Wu J.L and Sakai Y (1998), A Spatial Position Sensor for a Virtual Catch Ball System, Proceedings of the 3rd International conference on Advanced Mechatronics(ICAM'98), Okayama Japan, Aug 3-6, 1998, Vol.1 pp.139-144
4. Murata A, Zheng J.Yu, Fukagawa Y and Abe N(1997), Shape Recovery of Specular Objects from Multiple Lights ( in Japanese ), Transaction of Institute of Electronics, Information and Communication Engineers Vol. J80-D- II, NO. 7, 1997, pp.1659-1667

# Human Characteristics of Visual and Tactual Distance Perception on the Front Parallel-Plane for Teleoperation Systems

Toshio MIYAKE, †Jing-Long WU and Xiu-Ya LEI

Dept. of Mechanical Eng., Yamaguchi University  
Tokiwadai 2557, Ube, Yamaguchi 755-8611, Japan

† Dept. of Mechanical Eng., Faculty of Eng., Yamaguchi University  
Tokiwadai 2557, Ube, Yamaguchi 755-8611, Japan  
Email : wu@mechgw.mech.yamaguchi-u.ac.jp

## Abstract

*The haptic information is very important to develop a virtual teleoperation system with high reality. But it is very difficult to design a tactile device, in order to present haptic information. Because the human perceptive spatial scale of the real world is different from the virtual world. In order to investigate the relative characteristics of the spatial scale between the virtual and the real world, the characteristics of human visual and tactile distance perception on the horizontal plane were measured, in our pervious study. In this study, the characteristics of human visual and tactile distance perception on the front parallel-plane are measured. The results suggested that the human perceptive distances are larger (smaller) than the actual distances, if with using (not using) tactile information. In the case of using the visual information, the experimental results have individual differences. The experimental results provide useful basic data to design a tactile device, and to investigate the mechanism of distance perception.*

**Key words :** Human visual and haptic perception, Distance perception, Teleoperation system, Virtual reality.

## 1. Introduction

In pervious studies, the virtual reality has been mainly developed for visual applications[1~3]. In the virtual world, the haptic information is very important in order to operate a teloperator. But it is very difficult to design a tactile device. Because the human perceptive spatial scale of the real world is different from the virtual world.

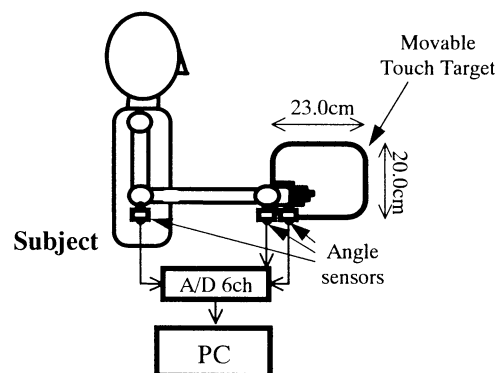
In our pervious study[4], we have measured the characteristics of the human visual and the tactile distance perception on the horizontal plane. The experimental results suggested that the perceptive distances were smaller than the actual distances if not using the tactile information, while the perceptive distances were larger than the actual distances if were using the tactile information. In the case of using the visual information, the difference between the actual and the perceptive distances were smaller than in the case of not using the visual information. In order to operate a teleoperator, many different directional

movements are needed to be operated. Therefore, the human characteristics of the visual and the tactile distance perception are needed to be measured on different directions. In this study, the characteristics of human visual and tactile distance perception on the front parallel-plane are measured. The experimental results suggested that the human perceptive distances are different from the actual distance. This experimental results provide useful basic data to design the tactile device, and to investigate the mechanism of human distance perception.

## 2. Psychological Experiments

### 2.1 Method of Experiments

In order to measure the characteristics of human distance perception, an experimental system was developed as shown in Fig. 1. The tactile information was presented by two movable touch targets for the left and the right hand, respectively. The touch targets were made from plywood, and they were wrapped by a black paper. Depth, height and width of the tactile targets were set at 23.0cm, 20.0cm and 0.5cm, respectively. The actual distance between the tips of the right and the left hand was measured by a standard scale and/or arm angles sensors.



**Fig. 1** The experimental system is used to measure the human distance perception on the front parallel-plane.

The concept of the arm angle sensor is explained in Fig.2.  $X$  is defined as the measurement distance between the tips of the left and right hand.  $E$  shown the

length from the right elbow joint to the left elbow joint.  $A$ ,  $B$  and  $C$  show the lengths from the elbow joint to the wrist joint, from the wrist joint to the finger joint, and from the finger joint to the tips of the fingers, respectively. The angles  $\alpha$ ,  $\beta$  and  $\gamma$  are defined in Fig. 2, and they are measured by three angle sensors.  $R$  and  $L$  show the right and left arm. Then, the distance  $X$  can be calculated by following equation.

$$X = E + \{A_R \sin(\alpha_R) - B_R \cos(\alpha_R + \beta_R - 180) - C_R \cos(\alpha_R + \beta_R + \gamma_R - 360)\} + \{A_L \sin(\alpha_L) - B_L \cos(\alpha_L + \beta_L - 180) - C_L \cos(\alpha_L + \beta_L + \gamma_L - 360)\} \quad (1)$$

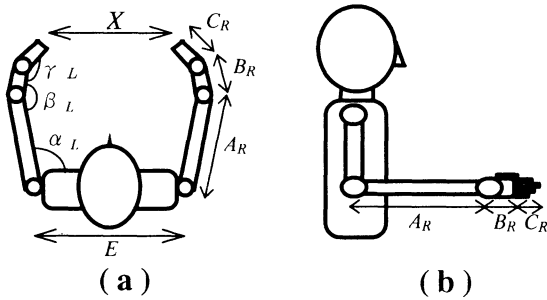


Fig. 2 The concept of the arm angle sensor

In order to define the subject's view size, a black blinder and a black frame are made for the side view and the front view, respectively. The frame and blinder are made from the same material which are wrapped by the black paper. The outline of the frame and blinder are shown in Figs. 3 and 4. The frame's length of depth, height and width are 35.0cm, 140.0cm and 98.0cm, respectively. The height of the black paper is 83.0cm. The distance between the frame and the subject, and the height of the chair are 50.0cm and 45.5cm, respectively.

The blinder's length of depth, height and width are 20.0cm, 18.0cm and 20.0cm, respectively.

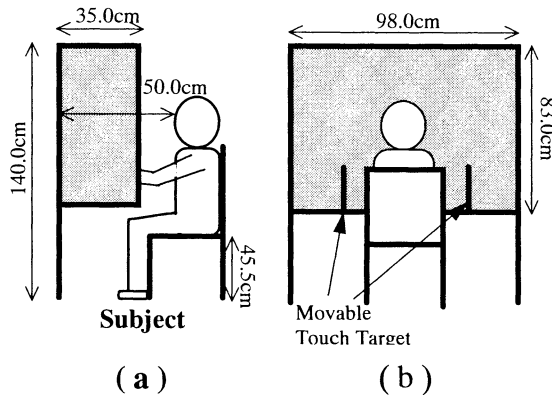


Fig. 3 The outline of the black frame

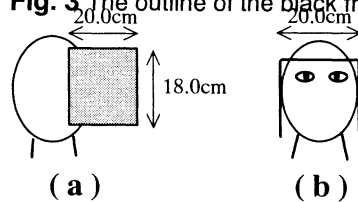


Fig. 4 The outline of the black blinder

## 2.2 Experimental Conditions

In four experiments, the subjects are requested to stretch his/her arms to present ( or perceive) a distance. The subject's two arms are concurrently stretched on the front parallel-plane. The actual distance and the human perceptive distance are measured concurrently. The experimental kinds are summarized in Table 1.

Table 1 Experimental kinds

Experiment No.	Using Touch	Using Vision
1	Yes	Yes
2	Yes	No
3	No	Yes
4	No	No

In Experiment 1, when a distance is set by the movable touch targets in a random order, the subject is requested to answer his/her perceptive distance with tactile and visual information concurrently. Namely, the subject stretch his/her two arms to touch the set movable touch targets with his/her finger tips, and answer his/her perceptive distance between the right and the left hand. In Experiment 1, the subject's sight is covered by the black frame and the black blinder.

Experiment 2 is similar to Experiment 1, but in order to answer a set distance, the subject used the touch but didn't used the vision. In Experiment 2, the subject's sight is blinded by a blindfold.

In Experiment 3, the subject is requested to stretch his/her arm in order to present some distance with vision but didn't present tactile information. Two arms angle sensors are worn on the left and the right arm of the subject. The subject is requested to present distance in a random order. For example, if "please present 30cm" is requested the subject stretch his/her arms of the left and the right to present this distance(30cm). In this case, the perceptive distance  $X$  is measured by the arm angle sensors (calculated by equation (1)), the actual distance is 30cm.

Experiment 4 is similar to Experiment 3, but in order to present a requested distance, the subject didn't use tactile and visual information.

In all the experiments, the measurement ranges of the distance are from 5 to 70cm. One distance is presented twenty eight times in one trial, and one trial is done five times for one subject.

## 2.3 Subjects

Five senior studies were served as subjects for all the experiments. The conditions of five subjects are summarized in Table 2. All subjects are men in their twenties.

## 3. Experimental Results

The results of all the Experiments of subject HY are shown here. The results of other subjects have similar characteristics.

### 3.1 Experiments 1 and 2

The results of Experiments 1 and 2 of subject HY are shown in Fig. 5. The horizontal and the vertical axes show the actual and the perceptive distance, respectively. The symbols  $\blacktriangle$  and  $\bullet$  show the perceptive distance of Experiment 1 (with visual and touch) and Experiment 2 (with touch but without vision), respectively. The dotted lines shown that the perceptive distance is equal to the actual distance. As shown in Fig. 5, if with tactile information, the perceptive distances are almost larger than the actual distances, regardless of using the visual information or not.

**Table 2** Subjects Conditions

Sub.	Arm	A	B	C	E	DA	VisualAcuity		DE
HS	Right	22.5	6.8	10.3	43.3	Right	Left	Right	Left
	Left	23.5	5.7	11.5			0.7	0.7	
HY	Right	23.5	6.7	11.3	43.3	Right	1.2	1.2	Right
	Left	24.2	6.5	11.2					
KS	Right	22.7	6.5	9.1	42.5	Right	1.2	1.2	Right
	Left	23.3	5.7	9.1					
ON	Right	23.7	6.9	11.0	42.4	Right	1.2	1.2	Left
	Left	23.6	6.5	12.0					
TM	Right	26.0	5.8	10.3	43.0	Right	0.5	1.2	Right
	Left	25.8	5.8	10.3					

Sub.: Subject, **A**, **B**, **C** and **E** are shown in equation (1), all units are the centimeter, **DA**: Dominant Arm, **DE**: Dominant Eye.

### 3.2 Experiments 3 and 4

The results of Experiments 3 and 4 of subject HY are shown in Fig. 6. The horizontal and the vertical axes show the actual and the perceptive distance, respectively. The symbols  $\triangle$  and  $\circ$  show the perceptive distance of Experiment 3 (with visual but without touch) and Experiment 4 (without touch and vision), respectively. The dotted lines are similar to in the case of Fig. 5. As shown in Fig. 6, if without tactile information, the perceptive distances are almost smaller than the actual distances, regardless of using the visual information or not.

## 4. Discussion

### 4.1 Perceptive and Actual Distance

In order to investigate the detail of the results of Experiments 1~4, a difference between the perceptive and the actual distance ( $PA$ ) is defined by the following:

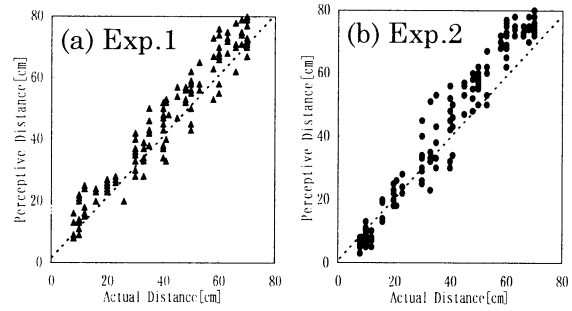
$$PA = PD - AD \quad (2)$$

where  $PD$  and  $AD$  are the perceptive and the actual distance, respectively.

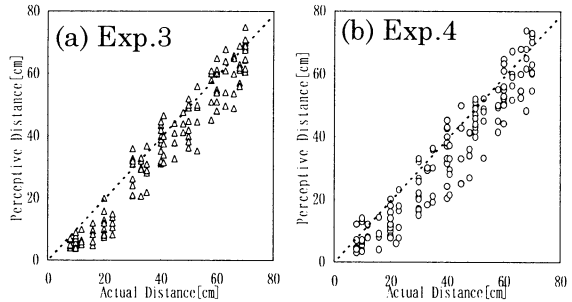
A ratio ( $R$ ) of the difference between the perceptive and the actual distance to the actual distance is defined by the following:

$$R = (PA / AD) \times 100 \quad (3)$$

The average values of the difference between the perceptive and the actual distance ( $PA$ ) and the ratio ( $R$ ) of subject HY are calculated from the experimental result of Figs. 5 and 6, and they are shown in Fig. 7. The horizontal axes show the actual distance. The vertical axes show the difference between the perceptive and the actual distance (Fig. 7(a)), and the ratio of the difference between the perceptive and the actual distance to the actual distance (Fig. 7(b)). The symbols  $\blacktriangle$ ,  $\bullet$ ,  $\triangle$  and  $\circ$  show the results of the Experiments 1, 2 3 and 4, respectively.



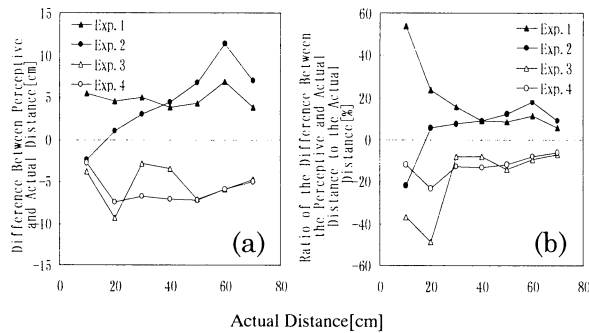
**Fig. 5** The results of Experiments 1 and 2 of the subject HY



**Fig.6** The results of Experiments 3 and 4 of the subject HY

As shown in Fig. 7(a), if using the tactile information (Exps. 1 and 2), the perceptive distance is almost larger than the actual distance ( $PA > 0$ ). But, if without tactile information (Exps. 3 and 4), the perceptive distance is smaller than the actual distance ( $PA < 0$ ). In the horizontal plane distance perception, the same phenomenon was confirmed[4]. This experimental results suggested that the tactile information is an influence on the human characteristics of large scale distance perception.

As shown in Fig. 7(b), the absolute value of the ratio ( $R$ ) is decreased as the actual distance is increasing, and it is nearly constant in the range of large actual distance. This experimental results suggested that the humans have high distance perceptive sensitivity in the range of the large distance. The experimental results of other subjects have similar tendency of Fig. 7.



**Fig. 7** (a) The difference between the perceptive and the actual distance(PA) and (b) the ratio(R) of the differences between the perceptive and the actual distance to the actual distance, as a function of the actual distance

The absolute value of all the subject's  $R$  are calculated from the experimental results of Experiments 1~4, they are summarized in Table 3.

**Table 3** The absolute value of the ratio ( $R$ ) of the differences between the perceptive and the actual distance to the actual distance in percentage[%] for all subjects

$R$ / Exp.	1	2	3	4
Sub.				
HS	32.8	10.5	20.7	23.6
HY	18.2	11.9	18.9	12.3
KS	9.62	12.9	6.29	12.5
ON	3.18	8.52	8.07	12.9
TM	6.32	10.3	4.85	8.07

Sub. : Subject, Exp. : Experiment, All unite of  $R$  are the percentage.

As shown in Table 3, the human distance perceptive characteristics have individual differences. In the case of using the visual information (Exps 1 and 3), the  $R$  of subjects ON and TM, are smaller than in the case of not using the visual information (Exps 2 and 4), regardless of using or not using the tactile information. In the case of using the visual information, the  $R$  of subject HY is larger than in the case of not using the visual information, regardless of using the tactile information or not.

In our pervious study[4], the experimental results of the horizontal plane suggested that the most highest accuracy of the human distance perception was using the tactile and visual information. The results of subject TM are same results of the horizontal plane, but the results of subject HY are different from the results of the horizontal plane. This results suggest that the visual information can advance or retreat on humans to perceive distance. Comparison between the experiments of using the tactile information (Exps 1 and 3) and experiments of not using the tactile information (Exps 2 and 4), the difference between the  $R$  of Experiments 1

and 3 is about equal to the difference between the  $R$  of Experiments 2 and 4.

Based on the experimental results, the distance information of the real world is not needed to be set equal to the distance of the virtual world, in order to design the tactile device. Because human perceptive distance is different from the actual distance. To propose a mathematical model of the distance perception is our future work.

## 5. Conclusion

In this study, we have measured the characteristics of human visual and tactile distance perception on the front parallel-plane. The experimental results suggested that the human visual and tactile distance perception have following main characteristics :

- (1) Same as in the case of the horizontal plane, humans have illusory characteristics of distance perception on front parallel-plane. Namely, the human perceptive distances are different from the actual distance.
- (2) If with using tactile information, the human perceptive distance is larger than the actual distance, but if without tactile information, the human perceptive distance is smaller than the actual distance, regardless of using the visual information or not.
- (3) The experimental results of the distance perception have individual difference. Therefore, to design the tactile device the individual difference is needed to be considered.

Based on the experimental results, to elucidate human mechanism of distance perception, and to design a tactile device are author's future work.

## References

- [1] S. S. Fisher, M. McGreevy, J. Humphries and W. Robinett : "Virtual Environment Display" . *ACM Workshop on Interaction 3D Graphics*, pp.23/24 (1986)
- [2] Jing-Long Wu, M. Nakahata and S. Kawamura : "A New Head Mounted Display System with Adjustable disparity for High Depth-Performance" . *IEEE International Conference on Systems, Man and Cybernetics*. **Vol.1**, pp.298/303 (1995)
- [3] Jing-Long Wu, M. Nakahata and S. Kawamura, I. Nishikawa and H. Tokumaru, : "Measurement of Binocular Stereoacuity for Design of Head Mounted Display System with Wide View" , *IEEE International Conference on Systems, Man and Cybernetics*. **Vol.2**, pp.929/934 (1996)
- [4] Jing-Long Wu and Toshio Miyake, : "Human Visual and Haptic Characteristics on Distance Perception for Virtual Teleoperation Systems" , *Proc. of Third Int. Symp. on Artificial Life, and Robotics (AROB III '98)*, **Vol.2**, pp719/722 (1998)

# Human Characteristics on Visual and Accelerative Perception for Virtual Simulator

Jing-long Wu and †Tsuyoshi Kiyooka

Dept. of Mechanical Eng., Faculty of Eng., Yamaguchi University  
Tokiwadai 2557, Ube 755-8611, Japan

Email: wu@mechgw.mech.yamaguchi-u.ac.jp

† Dept. of Mechanical Eng., Yamaguchi University  
Tokiwadai 2557, Ube 755-8611, Japan

**Key words:** *virtual reality, virtual simulator, cognitive science, human visual and accelerative perception*

## Abstract

In order to develop a virtual simulator with high performance, the information of the five senses are needed to be presented for the user. On the other hand, it is known that human senses have illusory characteristics. Therefore, to study interactive characteristic of the five senses is very important the purpose of developing the virtual simulator with low cost and high performance.

In this study, we aimed at the characteristics between the visual and accelerative senses. Up to now, it is reported that lot's of studies about the characteristics of equilibrium perception have been made. But, it is little about the characteristics between visual and accelerative perception. So the authors developed a motion sense presentation system, to measure the interactive characteristics between the visual and the accelerative perception. The experimental results suggested that the accelerative perception threshold is increased as the swaying frequency rises, however, the accelerative perception threshold is not remarkably changed if the visual information is presented or not.

## 1. Introduction

In the virtual simulator, the human sense of sight is the most important of the five senses[1]. We usually experience that any senses other than sight are caused by visual information. And it is known the illusion of the motion sense is caused by the sense of the sight. It is been studied the characteristics between visual and equilibrium sense, for example the influence of the rotation perception and posture control are caused by visual information and so on[2], [3]. The acceleration perception differed by the binding posture condition of the subject and the presentation of compound motion[4].

The goal of our study is to develop a low cost and high reality-performance virtual simulators using compound senses. In this study, the authors developed a motion sense presentation system with low cost for the virtual simulator, and measure the accelerative perception thresholds for the cases with visual information or not. The experimental results provide useful basic

data to design a virtual simulator with low cost and high performance, and to investigate the human mechanism of the visual and the accelerative perception.

## 2. Experimental system

### 2.1 Configuration of system

In order to measure the characteristics between the visual and the accelerative perception, an experimental system of the motion sense presentation was developed. Figures 1 and 2 are the configuration and the view of the experimental system, respectively. A moving chair was placed on the rails which were placed on the floor, and it was connected to a drawing motor (DC motor) by a wire. And the motor was connected an encoder to measure to the movement of the moving chair (see Figure 2(b)). The chair was able to move to the front and the rear. And also, the upper part of the moving chair was able to rotate. Figure 2(c) shows the rotation part. With using the moving chair, the subject could be moved to all directions in the horizontal plane. The subject was sat on the moving chair, he/she received the visual and the accelerative information from a Head Mounted Display (HMD) and the moving chair, respectively. Visual information was played by a videotape recording (CCD video camera).

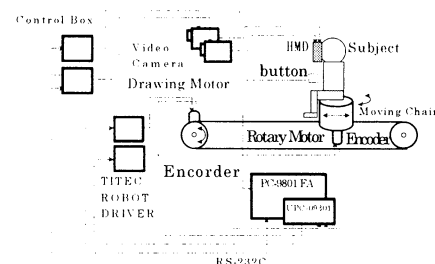


Figure 1 The configuration of the experimental system

In order to get the synchronization of the movements of two video cameras with the moving chair, two video cameras were concurrently controlled by a video/computer interface unit (SONY CI-1100). The reaction of the subject was taken in personal computer through a reaction button.



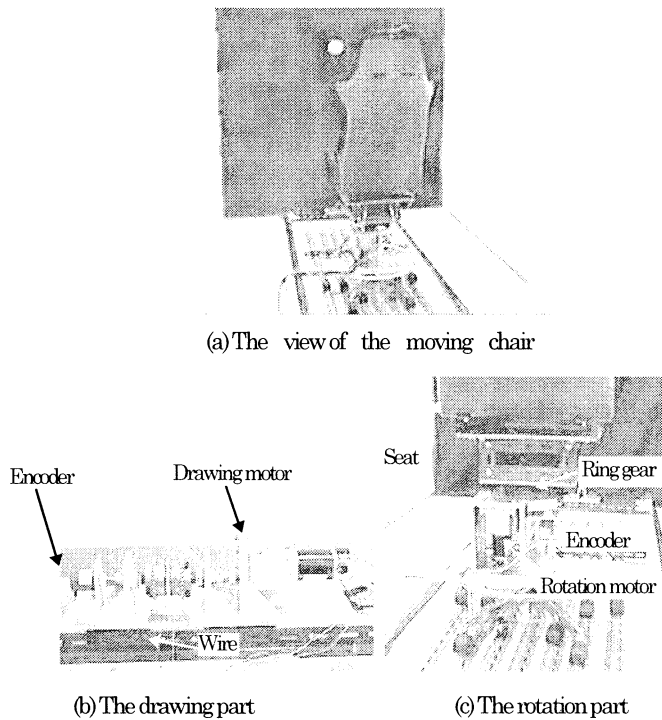


Figure 2 The view of the system

## 2.2 Basic performances of the system

Maximum moving velocity of this system is about 13cm/s, and maximum movement distance is about 3.5m. The movement of the moving chair was measured by an encoder (the resolution is 1000 Counts/Turn), and the sampling frequency was about 16Hz (it is enough to measure the reaction of the subject).

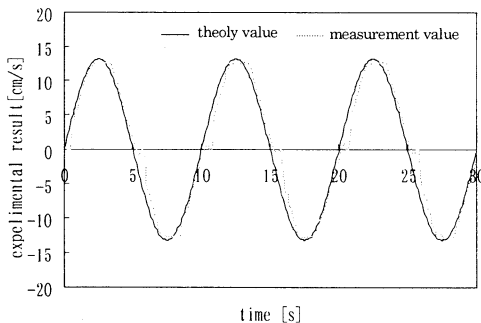


Figure 3 The dynamic characteristics of the drawing part

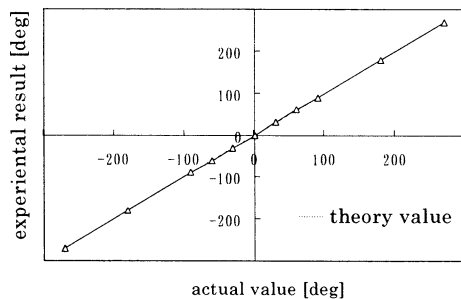


Figure 4 The static characteristics of the rotation part

The measurement resolution of the motion was 0.044cm/s at velocity, or 0.75cm/s<sup>2</sup> at acceleration. Figure 3 shows the dynamic characteristics of the drawing part. Figure 4 shows the static characteristics of the rotation part. The results were measured on the condition of that the load was set at 60Kg weight.

## 3. Experimental method

### 3.1 Motion stimulus

Motion stimuli were the front and the rear swaying like the corrugation which is shown in Figure 5. The motion of the velocity was as 2<sup>nd</sup> curved line, so the motion of the acceleration was as 1<sup>st</sup> straight line (in other words, the acceleration motion ratio was constant). One swaying cycle is defined that the moving chair goes and returns for the front and the rear once. In this study, the swaying frequency was set at 0.1000, 0.0625, 0.0410 or 0.0200Hz. The frequency, the maximum displacement, and the maximum acceleration are shown in Table 1.

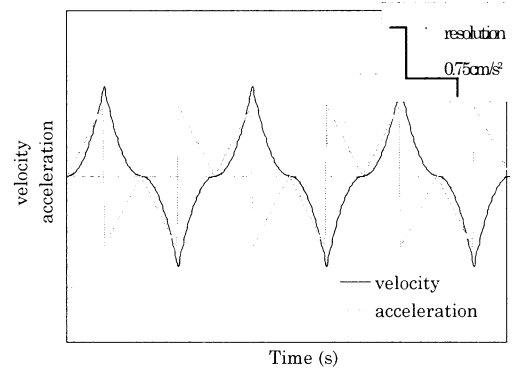


Figure 5 The motion velocity and the acceleration of the motion stimulus.

Table 1 The maximum displacement and acceleration.

swaying frequency [Hz]	maximum displacement [cm]	maximum acceleration [ cm/s <sup>2</sup> ]
0.1000	19.6	9.7
0.0625	31.3	6.7
0.0410	47.4	4.5
0.0200	97.3	2.3

### 3.2 Visual stimulus

The visual conditions were the cases of that the visual information was not presented (case 1), and the case of that the visual information was presented (case 2). In the case 1, the subject's sight was blinded by a blindfold, and in the case 2, the visual information was presented by the HMD the images were taken in the real world beforehand. The images were taken by video cameras which fixed on the height of the subject's eye line in a model room as showed in Figure 6. A white ball (diameter 10cm) was used as a visual fixation point. The image of the visual stimulus and the motion stimulus are shown in Figure 7.

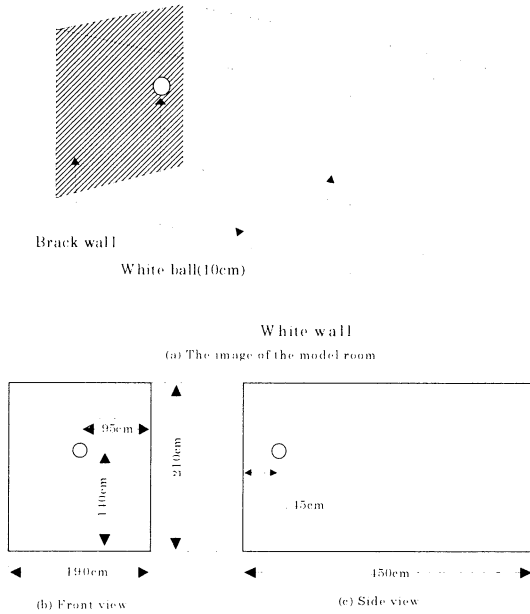


Figure 6 The model room

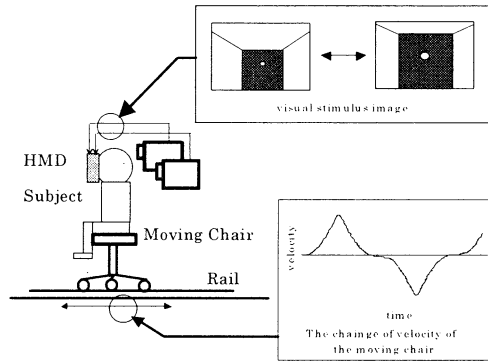


Figure 7 The images of the experimental stimuli

### 3.3 Subjects

Subjects were four males who were 21–24 years old, and they had no problem on the balancing sense and the sight sense organs. Their weight were between 50 and 65Kg.

### 3.4 Experimental procedure

The subjects were sat on the moving chair, and using the reaction button to answer his/her velocity sense. When the velocity of the visual or/and the motion stimuli were changed, if the subject was sensing that the velocity was accelerated, he/she pushed the reaction button. At this moment, the absolute value of the acceleration was memorized, and this value is defined as the accelerative perception threshold of the rising process. Conversely, if the subject was sensing that the velocity was descended, he/she released the reaction button. At this moment, the absolute value of the acceleration was memorized, and it is defined as the accelerative perception threshold of the descending process.

The visual stimuli were presented for the subject that had taken in the same frequency as the swaying frequency of the

motion stimuli. One try is defined, as that the moving chair goes and returns alternately for the front and the rear, and one session is defined as three tries. The different swaying frequency was shown for the subject in a random order. The measurements were divided in the cases 1 and 2. For one subject, the measurements of the same condition were carried out on three sessions.

## 4. Results

The experimental results of four subjects are shown in Figure 8. Each datum of Figure 8 shows the average of 108 data of four subjects. Figures 8 (a) and (b) are the results of the front and the rear of accelerative perception threshold in the case 1 (no visual information), respectively. Figures 8 (c) and (d) are the results of the front and the rear of accelerative perception threshold in the case 2 (using visual information), respectively. The black lines and the gray lines show the results of the rising and the descending processes, respectively. The vertical bars show the change ranges of the data in the different measurement try.

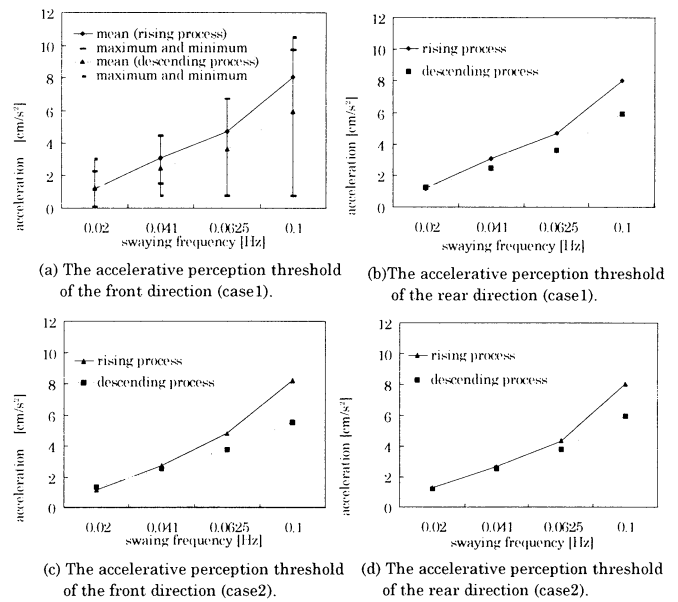
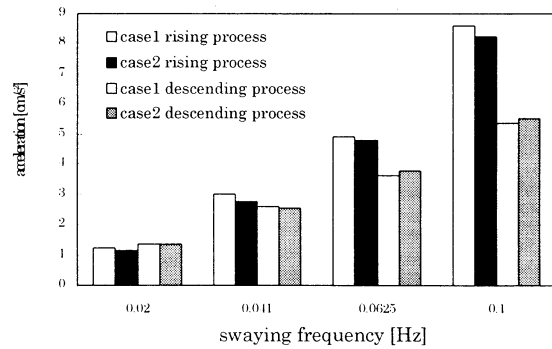
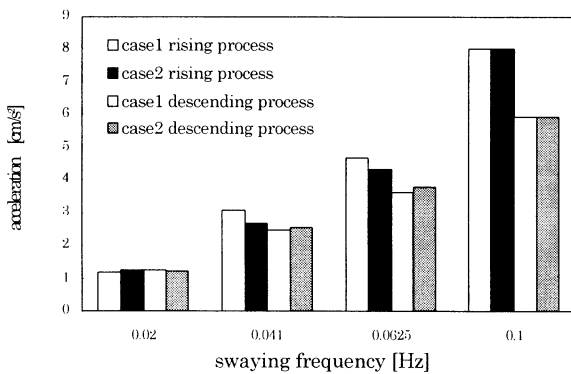


Figure 8 The accelerative perception threshold of each swaying frequency

From Figure 8, it can be seen that the threshold increases as the swaying frequency is increased. And also, the differences of the threshold between the rising process and the descending process have larger value in the range of high frequency. The thresholds are not remarkably changed between the cases of the front and the rear directions. Figure 9 shows the comparison between experimental results of the case 1 and the case 2. From Figure 9, it can be seen that the threshold of the rising process is larger than the descending process's one. However, the accelerative perception thresholds have not systematically changed between in the case 1 and in the case 2.



(a) The accelerative perception threshold of the front direction.



(b) The accelerative perception threshold of the rear direction.

Figure 9 The comparison between the experimental results of the case 1 and the case 2

## 5. Discussion

From the experimental results, we can summarize the human visual and accelerative characteristics as following:

- (1) The accelerative perception threshold increases as the swaying frequency increases from 0.02 to 0.1Hz. That is to say, humans discriminate the change of the high speed motion that is difficult than the case of low speed motion.
- (2) The accelerative perception threshold of the rising process is larger than the descending case's value. In our future work, it is needed to investigate in detail.
- (3) Our results have not remarkable differences of the accelerative perception threshold between the case 1 (not using visual information) and the case 2 (using visual information). We consider the reason that the view field of the visual information was small ( $25.5 \times 22.3$  degree), and the depth information was extremely little in the visual stimulus because the side walls of the model room were white uniform planes. Therefore, we can say that the visual information is not remarkably effected to the accelerative perception, if the view field size and the depth of the visual stimulus are set at a smaller values. This results suggested that if we want to simplify the structure of the virtual simulator with using visual information, the wide view field

and the depth information of the visual stimuli are needed to be presented. Previous study suggested that the peripheral view is very important in the motion perception[5]. It is our future work to discover a quantitative relationship between the view filed size and the depth information of the visual stimulus, and the accelerative perception.

## 6. Conclusion

In this study, the motion sense presentation system with low cost was developed, and the basic performances of the system were demonstrated. Using the proposed system, accelerative perception thresholds have been measured. The experimental results suggested that the accelerative perception was effected by the swaying frequency, and was not remarkably effected by the visual information if the view field and the depth information were set at smaller values. Based on this results, we have known that in order to simplify the structure of the virtual simulator with using visual information, the wide view field and the depth information of the visual stimuli are needed to be presented.

In this study, the resolutions of the acceleration measurement were about 10-30%. In the future, we are going to make the resolution of this system improvement.

## Acknowledgments

The authors would like to express our gratitude to the subjects. The authors are also grateful to Mr. Masayuki Kitazawa for his help in developing of the experimental system.

## References

- [1] T. Sawada, I. Imanishi and Y. Oguchi (1996), Influence of human sense on simulator display (in Japanese). 6<sup>th</sup> proc. of Design and Systems (JSME), pp.166-168
- [2] H. Nara, M. Takahashi and T. Ifukube (1996), Effects of moving visual stimulation on the sense of equilibrium (in Japanese). 11<sup>th</sup> proc. of Symposium on Biological and Physiological Engineering, pp.25-28
- [3] H. Nakamura, N. Uemi, H. Nara, T. Tanaka and T. Ifukube (1997), Influence of light or sound stimulation on the perception of rotatory motion (in Japanese). TECHNICAL REPORT OF IEICE HIP96-46: pp.43-46
- [4] T. Ohyama, S. Imai and T. Waki (1994), New edition Handbook of Sensation perception psychology (in Japanese). Seishinnsyobo
- [5] H. Nara, M. Takahashi, Y. Ishida, M. Kimura, K. Mitobe and T. Ifukube (1995), Design of a new Head Mounted Display using peripheral vision and its effects on postural control (in Japanese). TECNICAL REPORT OF IEICE MBE94-134: pp.87-94

# Human Visual and Auditory Characteristic in the Temporal Frequency Domain

Jing-Long Wu and Osamu Nobuki

Department of Mechanical Engineering, Yamaguchi University  
Tokiwadai 2557, Ube, Yamaguchi 755-8611, Japan  
Email:wu@mechgw.mech.yamaguchi-u.ac.jp

## Abstract

Recently, many audio-visual devices have been developed to the expansion field of the virtual-reality and the diversification of the mass media. In these applications, the visual and the auditory information is very important. Because humans mainly get the environmental information from the visual and the auditory senses. In order to get high reality-performance, the human audio-visual characteristics and mechanism are needed to be investigated. In this study, the phase discrimination thresholds between the visual and the auditory stimulus are measured with the changing of the temporal frequency of the stimuli. The experimental results suggested that the phase discrimination thresholds are increased as the temporal frequency of the stimuli is increasing. The experimental results are possible to apply effective productions in the virtual-reality, the drama and the mass media, they can contribute to elucidate the human audio-visual mechanism.

## 1. Introduction

In previous studies, human visual and auditory characteristics have been separately investigated in detail[1]. There are studies that argued visual or auditory characteristic each come to elucidate the channel of information extended to the brain[2]. The research about the audio-visual mutuality's related characteristics in spatial domain are to see an example of the sound source localization, is examined well[3]. In the application fields of the mass media and the virtual reality, the effect of the matching of sound and video were investigated [4][5]. There are studies discussed about the audio-visual mutuality's related characteristic in recognition science[6][7]. However, there are few things which argued in the temporal domain about the audio-visual mutuality's related characteristic.

In this study, we pay attention to characteristic of the human audio-visual information processing. The phase discrimination thresholds between the visual and the auditory stimulus are

measured with changing the temporal frequency of the stimuli. The experimental results suggested how to present the visual and the auditory information concurrently in the virtual environments, and they provided basic data to clarify human mechanism of the visual and the auditory information processing.

## 2. Experiment

### 2.1 Experimental System

A proposed experimental system is shown in Fig. 1. The visual stimuli were generated by a personal computer (NEC PC9821), and presented by a CRT (Sony CPD-17SF7). The audio stimuli were generated by the personal computer, and presented by a headphone (Sony MDR-CD370).

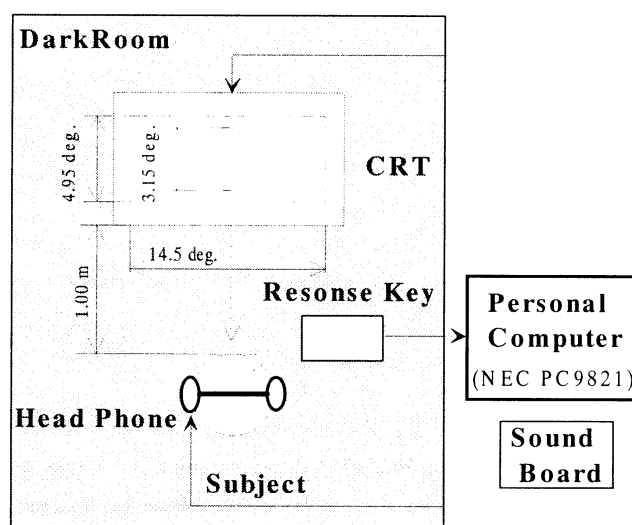


Fig. 1 The outline of the experiment system

### 2.2 Experimental Method

The visual experimental stimuli are shown in Fig. 1. The distance between the CRT and the subjects is set at 1.00m. The flame size of the rectangle background is  $4.95^\circ \times 14.5^\circ$ , and it's brightness is set at  $25.8 \text{ cd/m}^2$ . The diameter of the circle is set at

3.15°, as visual target stimulus. The location of the target stimulus is in the center of the rectangle background. The brightness(gray scale) of the target stimulus is varied between 0.970 cd/m<sup>2</sup> and 52.4 cd/m<sup>2</sup> in a sine curve. So, the target visual stimulus(the circle) is changed by brightness only.

The pitch of the auditory stimuli is set at 1 kHz or 2 kHz. The loudness of the auditory stimuli is varied between 48.6 dB and 58.0 dB, 48.1 dB and 63.5 dB, in a sine curve of the time course. In order to make the exact sine curves of the visual and the auditory stimulus in the time course, the CRT brightness and the sound pressure level were calibrated. The measurements are carried out in the conditions of that the temporal frequency is set at 0.18, 0.32, 0.56, 0.98, 1.7 and 3.0 Hz. In the same condition, the measurements are carried out in the cases of that the phase differentiates between the visual and the auditory stimulus set at 0°, 30°, 60°, 90°, 120°, 150° and 180°. The phase difference means that the phase value of the visual stimulus precedes auditory one. This reason is for the consideration of the subjects' health. The scheme diagram of the visual and the auditory stimulus is shown in Fig. 2.

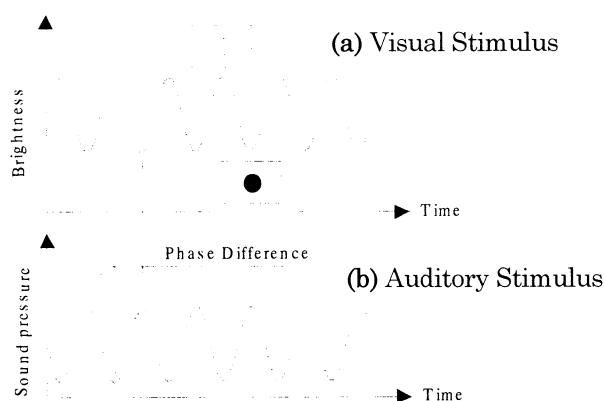


Fig. 2 The schematic diagram of visual and auditory stimulus on the time course

When the visual and the auditory stimulus are presented concurrently(Fig. 2), the subjects are required to press the response key to indicate whether between the peak of brightness and the peak of sound pressure of stimuli have been recognized synchronized or not. That is to say, if the subjects sense that the visual stimulus is in(out) phase with the auditory stimulus, he/she presses the response key to answer "Yes" ("No") when the different types of the visual and the auditory stimulus are presented. The time chart of the measurements is shown in Fig. 3. The combinations of changing the temporal frequency and the phase of the visual and the auditory stimulus, are 42 per a pitch

of the auditory stimulus. Each combinatorial stimulus is presented 16 times in a random order. Three students were served as the subjects for all the experiments. The conditions of the three subjects are shown in Table 1.

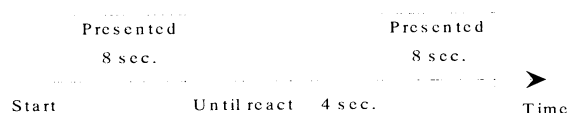


Fig. 3 The time chart of the measurements

Table 1 Subject Conditions

Subject	Age	Sex	Visual Acuity	Hearing Ability
H.Y	22	Male	R:1.2, L:1.5	Normal
T.K	22	Male	R:1.2, L:1.2	Normal
O.N	21	Male	R:1.2, L:1.2	Normal

### 3. Results

The experimental results are shown in Fig. 4. Figures 4 (1) and (2) are the results of that the pitches of auditory stimulus is set at 1 and 2 kHz. The vertical axis indicates the ratio of the subjects reply "Yes" ( the visual stimulus is in phase with the

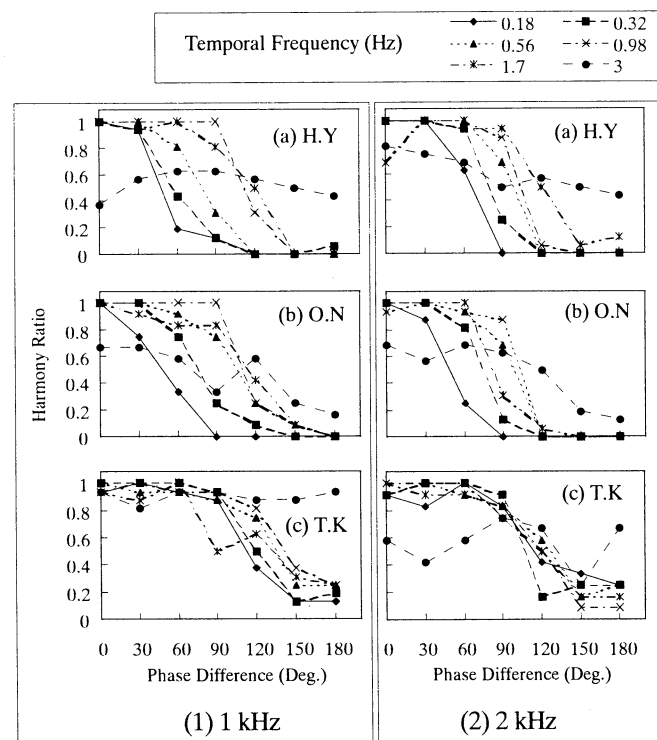


Fig. 4 The harmony ratio as a function of the phase difference.

auditory stimulus) times to the total times of that the target stimuli are presented. This ratio is defined as the harmony ratio.

From Fig. 4, it can be seen that the harmony ratio is decreased as the phase difference is increasing, except the case of that the temporal frequency is set at 3Hz. In the preparatory experiment, the harmony ratio can not be measured if the temporal frequency is larger than 3 Hz. Because the visual and the auditory stimulus is too fast, the subjects can not discriminate that the visual stimulus is in phase with the auditory stimulus or not. So we can say that human can not discriminate the phase difference between the visual and the auditory stimulus if the temporal frequency is larger than 3 Hz.

As shown in Fig. 4, we can detect that the decreasing of the harmony ratio depends on the temporal frequency. This peculiarity can be explained by using Fig. 5. As shown in Fig. 5, when the phase difference is increased, the harmony ratio decreases quickly in the case of the high temporal frequency, conversely, the harmony ratio decreases slowly in the case of the low temporal frequency.

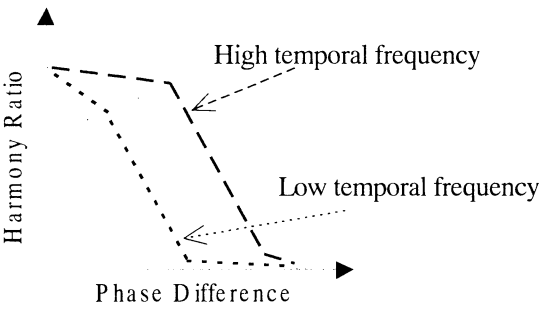


Fig. 5 Schematic tendency of characteristic between harmony ratio and phase difference

#### 4. Discussion

Figure 6 shows the characteristic between the harmony ratio and the temporal frequency. The Figs. 6(1) and (2) are results of that the pitch of auditory stimuli is set at 1 and 2 kHz, respectively.

From Fig. 6, we can see that the harmony ratio has different characteristics with changing the temporal frequency. In the cases of the small (0° and 30°) and the large(150° and 180°) phase differences, the harmony ratios are nearly constant. In the case of that the phase difference is 120°, the harmony ratio is increased as the temporal frequency is increasing. However, the harmony ratio has band pass characteristic in the cases of that the phase differences are set at 60 and 90°.

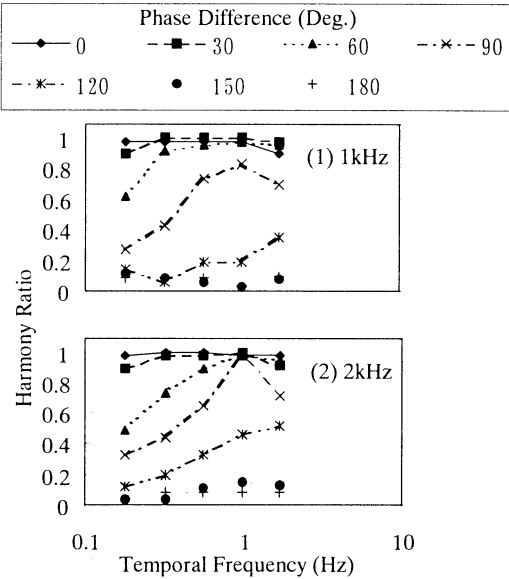


Fig. 6 The harmony ratio as a function of the temporal frequency

Figure 7 shows the average experimental results of the three subjects, it is replotted from Fig. 4. In order to investigate the detail of the characteristics of the harmony ratio, phase discrimination thresholds between the visual and the auditory stimulus can be calculated from Fig. 7. The phase discrimination threshold is defined with the phase difference at the harmony ratio value with 75%. The threshold is found at the cross point of the harmony ratio value and the 75% line, as shown in Fig. 7. In this way, the phase discrimination thresholds are calculated, and they are summarized in Table 2.

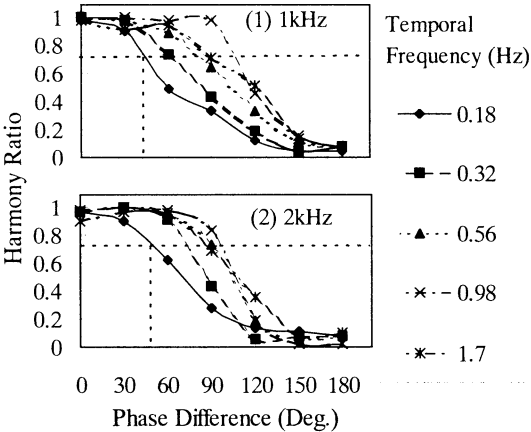


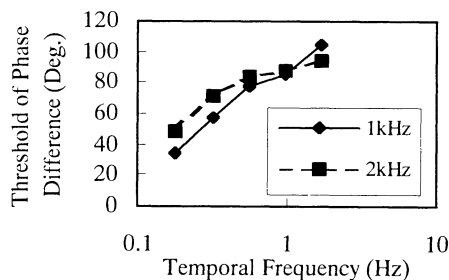
Fig. 7 Average harmony ratio of the three subjects as a function of the phase difference

Figure 8 shows the characteristics between the discrimination threshold of the phase difference and the temporal frequency. Figure 9 shows the characteristics between threshold of time lag

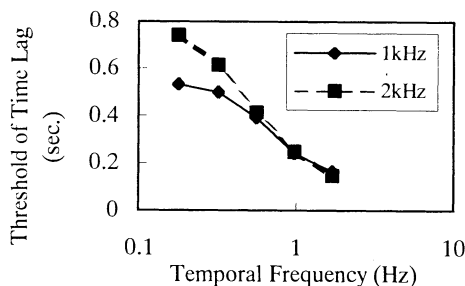
75% harmonized and the temporal frequency. The discrimination threshold of the time lag can be easily calculated from the discrimination threshold of the phase difference. Because the temporal frequency is a constant (0.18, 0.32, 0.56, 0.98 or 1.7Hz).

**Table 2** The discrimination threshold of the phase difference and the time lag

Sound pitch	Threshold of difference phase and time lag	Temporal frequency (Hz)				
		0.18	0.32	0.56	0.98	1.7
1kHz	Phase (deg.)	34.6	57.2	78.4	85.6	105
	Time lag (sec.)	0.53	0.5	0.39	0.24	0.17
2kHz	Phase (deg.)	47.9	71.3	84.3	88.6	94.7
	Time lag(sec.)	0.74	0.62	0.42	0.25	0.15



**Fig. 8** The characteristic between the discrimination threshold of the phase difference and the temporal frequency



**Fig. 9** The characteristic between the discrimination threshold of the time lag and the temporal frequency

In Fig. 8, the discrimination threshold of the phase difference is getting larger and larger as increasing of temporal frequency, the cases of 1 kHz and 2 kHz alike. On the other hand, seeing Fig. 9, the discrimination threshold of the time lag decrease as increasing of temporal frequency, 1 kHz and 2 kHz alike. It is considerate that the reason the tendency of 1 kHz differ from 2 kHz is that the human equal-loudness sensitively varies from the pitch of sound stimuli.

## 5. Conclusion

In order to investigate the human visual and auditory characteristic in the temporal frequency domain, the discrimination thresholds of the phase difference between the visual and the auditory stimulus, are measured. The discrimination threshold of the phase difference is getting larger and larger as increasing of the temporal frequency. And this characteristic varies from the pitch of the auditory stimuli. It is said that it is no longer able to distinguish of the phase difference as the temporal frequency is higher than 3 Hz.

From now on, the temporal frequency of presentational stimuli should be accurate, it should be measured the effect to discriminate the phase difference of the shape of visual stimuli and the pitch of the auditory stimuli, so it can be elucidated the mechanism of the audio-visual information. And investigating more minutely, it is possible to produce a more effective for virtual reality, drama and mass media.

## Acknowledgments

The authors would like to express our gratitude to the subjects. The authors are also grateful to Mr. Masayuki Kitazawa and Mr. Hiroaki Mizuhara for their help in the programming of the experiments.

## References

- [1]. Kunio Kashino and Hidehiko Tanaka, "A computational Model of Auditory Segmentation of Two Frequency Components - Evaluation and Integration of Multiple Cues-(in Japanese)", IEICE, Part A, Vol. J77-A, No. 5, pp. 731-740 (1994)
- [2]. Stephen Engel, Xuemei Zhang & Brian Wandell, "Colour tuning in human visual cortex measured with functional magnetic resonance imaging", Nature, Vol. 388, pp. 68-71, 3 July (1997)
- [3]. Michiko Ohkura, Yasuyuki Yanagida and Susumu Tachi, "Sound Distance Localization Using Environment(in Japanese)", SICE, Vol. 31, No. 9, pp. 1318-1323 (1995)
- [4]. Shin-ichiro Iwamiya, "The effect of the matching of sound and video on the interaction between auditory and visual processing in communication via audio-visual media(in Japanese)", JASJ, Vol. 48, No. 9, pp. 649-657 (1992)
- [5]. Shin-ichiro Iwamiya, "The interaction between auditory and visual processing when listening to music via audio-visual media(in Japanese)", JASJ, Vol. 48, No. 3, pp. 146-153 (1992)
- [6]. Jing-Long Wu, Hiroaki Mizuhara and Yoshikazu Nishikawa, "Measurement of Human Visual and Auditory Characteristics for A Virtual Driving System", Proc. of AROB 3rd, 98, Oita, Japan, 19-21, January, 1998
- [7]. Jing-Long Wu, Hiroaki Mizuhara and Yoshikazu Nnshikawa, "Characteristics of Reaction Time in Human Visual and Auditory Information Processing(in Japanese)", Memories of the faculty of Engineering Yamaguchi University, Vol. 49, No. 3, March 1999, be in press.

## Immune algorithm with immune network and Major Histocompatibility Complex

Naruaki Toma

tnal@eva.ie.u-ryukyu.ac.jp

Graduate School of Science and Engineering

University of the Ryukyus

1 senbaru, Nishihara, Okinawa 903-0213

Japan

Satoshi Endo , Koji Yamada

{endo,koji}@ie.u-ryukyu.ac.jp

Faculty of Engineering

University of the Ryukyus

1 senbaru, Nishihara, Okinawa 903-0213

Japan

### Abstract

Adaptive problem solving techniques such as neural networks and genetic algorithms become so popular in the AI field. The biological immune system is one of the adaptive biological systems whose functions are to identify and to eliminate foreign material. In this paper, we propose an adaptation algorithm based on immune model with immune network and Major Histocompatibility Complex. Immune network is a learning technique as an adaptive problem solving. MHC expresses a characteristic that an agent can act some behaviors. To investigate an adaptation ability of the proposed algorithm, we apply to n-th agent's travelling salesman problem called n-TSP. This algorithm performs adaptive behaviors for distributed cooperation.

## 1 Introduction

Such as neural networks and genetic algorithms, adaptive problem solving techniques become so popular. These techniques are based on information processing in biological organisms and are applied many kinds of optimization problems[1]. On the other hand, the biological immune system is widely recognized as one of the adaptive biological system whose functions are to identify and to eliminate foreign material. In order to perform such functions, Major Histocompatibility Complex (MHC) is used to identify foreign material by difference of MHC. Immune network is used to control making antibodies to eliminate antigens by activation among immune cells.

In this paper, we propose a model of immune functions and an immune algorithm as an adaptive problem solving on multi-agent system. Immune network is defined as controlling mechanism of immune response to eliminate effectively. MHC expresses a characteristic that an agent can act some behaviors and uses to cooperation among the agents. By using their char-

acteristics, this algorithm will perform effective adaptation in dynamically changing environments by the distributed cooperative among some agents. Then, we apply the proposed algorithm to n-th agent's travelling salesman problem (called n-TSP) which is one kind of multi-agent systems. Some computer simulations are designed to investigate the performance of the proposed immune algorithm.

## 2 Biological immune system and proposed algorithm

### 2.1 Immune network

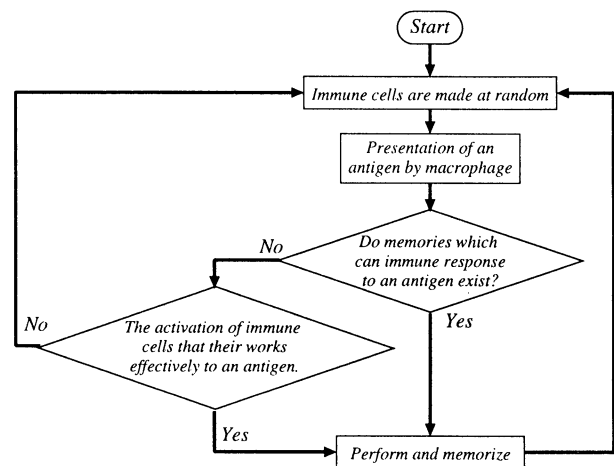


Figure 1: Memory mechanism with immune network

Immune network is defined as controlling mechanism of immune response, primary and secondary immune response, by activation on the network that is composed of immune cells and their connection. The immune cells are composed of following three cells. First one is macrophage whose function defined as primitive response (random recognition, presentation



and elimination against foreign materials). Second one is T cell that recognizes infected cells (activation of development of antibodies, recognition and elimination by matching between T cell antigen receptor(TcR) and foreign material). Third is B cell that recognizes uninfected cells (presentation and development of antibodies by matching between its antibody and foreign material). By mutual cooperation among their cells, especially essential functions are 1) widely recognition ability by 3 kinds of recognition, 2) strongly adaptation ability to unknown vast antigens by production of random solutions controlling of immune response, 3) memory mechanism is used to eliminate antigens that eliminated once.

In the multi-agent system, there is environments as dynamically changed by autonomic behavior of agents. The exhaustive search for the environments is difficult problem because the problem space is too large. The biological immune system will have the advantages of learning and adaptive technique on the environments because the system is possible to make solutions against unknown vast problems (i.e., the problems which immune systems deal with are too large like dynamically changing environments).

## 2.2 MHC

In the biological immune system, a characteristic of an individual is defined as MHC[2]. If the infected MHC on the cells which is composed an individual is different from original(uninfected) MHC, the individual can recognize antigens by matching with TcR. In other words, MHC is used to distinguish a "self" from other "not self".

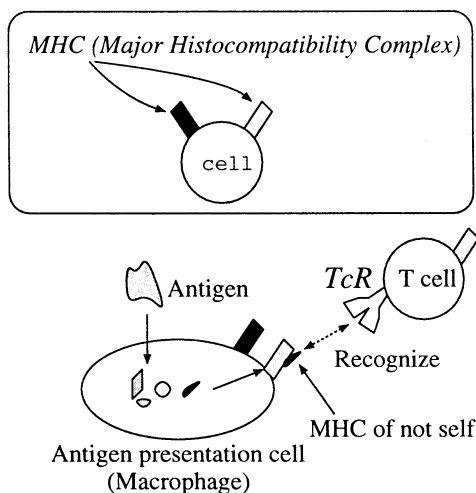


Figure 2: MHC and recognition of not self

In the design of effective multi-agent system, distributed operation with efficiency by mutual cooper-

ation among agents is one of the most important aspects. It is thought as a such subject that the operations which don't include repeat operation among agents design. In other words, we aim to construct the system whose operations of every agent compose exclusive ones. To consider the construction with biological immune system, agents own operations that are necessary information for establishment of "self" is defined as MHC, at first. By such definition, it is thought one aspect that MHC between agent A and agent B is difference, but a part of MHC both of agents is equal. In such a case, if both of agents recognize other agent as a "not self" and try to eliminate an opponent, the system that is composed such agents will hope to be the purposed construction.

## 2.3 Adaptations in biological immune system

It is thought of as two types of adaptation mechanism as a view, pre-adaptation and post-adaptation, in biology. It considers one case that the individuals change self-inertia to exist against selection pressure from dynamically changing environments. The pre-adaptation is defined as changing self-inertia in an aspect before selection pressure occurs (i.e., adapt independently of selection pressure). The post-adaptation is defined as changing self-inertia in an aspect after selection pressure occurs (i.e., adapt in consideration of selection pressure). When adaptation system to the environments is composed, is important how to implement their two types of adaptation mechanisms. In the biological immune system, immune network takes charge of them. Making of random solutions means the pre-adaptation and adaptation using memory means the post-adaptation. It will be possible to construct that an adaptation model keeps being effective adapted (i.e., without the condition that falls into the local optimum).

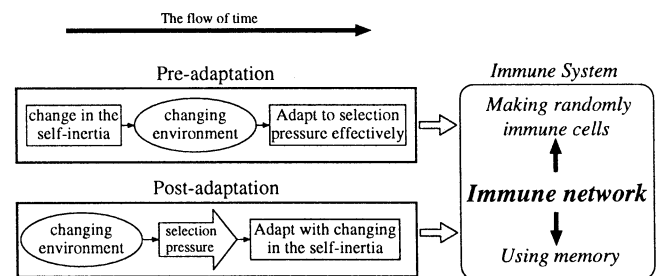


Figure 3: Pre-adaptation and post-adaptation in the immune system

## 2.4 Proposed immune algorithm

The proposed IA to adaptive solving for multi-agent model illustrated on figure 4. Each step of the proposed IA is explained in the following. At this point, antigens are solving problems of not self agents and environments, and one individual composed of immune cells is one agent.

### Step1. Definition of an antigen

Information which agents can recognize is defined.

### Step2. Own establishment

MHC of agents as each an individual's characteristics are defined, and initial agents composed of immune cells (here, macrophages, T cells, and B cells) make. The immune cells make with random or memory.

### Step3. Swapping the immune cells

The immune cells that their energies as a life-time was lost, become extinct. Make new cells with random and swap lost cells for new cells.

### Step4. Acquisition of the environment information and presentation to the agent

Agents can acquire information of not self by using macrophages. The macrophages that have information about not-self change own MHC itself.

### Step5. Energy transmission for activation

Agents recognize not self by T cells and B cells that are matching MHC about not self. When T cells and B cells that recognized it, they get energy from macrophages. The energy that each cell can get is in proportion to their affinity, match degree of between information of macrophages and T cells (or B cells).

### Step6. Memory and production of an antibody by division and differentiation

T cells and B cells that energy exceeded their threshold divide to memory cells and (only B cells) antibody producing cells. Memory cells continue to exist as a memory of recognized antigens. When agents meet with the same antigen memory cells, can recognize it, produce antibodies quickly than first encounter.

Four steps, from step 3 to step 6 are one time step. In addition, this algorithm is adapted by repeating from step 3 to step 6 until terminal condition.

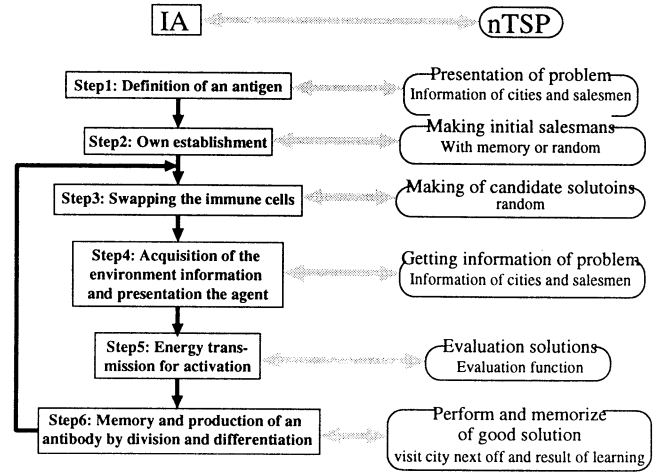


Figure 4: Proposed IA and n-TSP

## 3 Experiment

### 3.1 Design of IA for n-TSP

The design of antigens and immune cells are as the following.

[Antigen] : Information of “cities” and “salesman”.

- cities;
  - $m$ ; a number of cities.
  - $Pos(m)$ ; position of cities.
  - $Dis(m_i, m_j)$ ; distance between  $m_i$  and  $m_j$ .
- salesmen;
  - $n$ ; a number of salesmen.
  - $Now\_Town$ ; the city which it is in at this present.
  - $MHC[] = Traveled\_Town[1..m]$ ; set of visited city.
  - $Cost(Traveled\_Town[]) = \sum_i^m Dis(i, i+1)$ ; cost of traveled tour.
  - $Strength = 1/Cost()$ ; easiness to spread the tour.

where  $MHC$  is used as communication between agents. The evaluation value when an agent moves to the city  $m_{action}$ , as the following (i.e. an evaluation function in the step 5 of figure 4).

$$Eval(m_{action}) = Dis(Now\_Town, m_{action}) * Strength \quad (1)$$

The value becomes lower as much as the cost of the tour is long. Therefore, the tour whose cost is smaller is supposed to be designed by the equation.

[Macrophage] : Acquisition from an antigen. Composed of  $Traveled\_Town[]$ , “Local\_View[]” and “Energy”.

- *Local\_View*[1...*m*]; information of partial cities.
- *Energy*; life-time.

where *Local\_View*[ ] which gets from an antigen presents to the agent.

**[T cell] : Helping activation of B cell.**

Composed of *Local\_View*[ ] and *Energy*.

When the *Local\_View*[ ] of Macrophage and T cell's one match only, T cell help activation of B cell (i.e. step 5 in the algorithm).

**[B cell] : Making antibodies.**

Composed of "*m<sub>action</sub>*", *Local\_View*[ ] and *Energy*.

- *m<sub>action</sub>*; visit city next off.

When one B cell that was helped by T cell makes an antibody as a behavior.

**[Antibody] : Perform behavior of agent.**

Composed of *m<sub>action</sub>*.

### 3.2 Definition of experiment

The problem defined in the figure 5. The arrangement of the cities is a circle, which a start city was moved to the center, and the circumferential cities are being arranged uniformly. The performance is confirmed in this experiment with two standards. First is a visited all cities, and second is the cost proceed in the optimization direction.

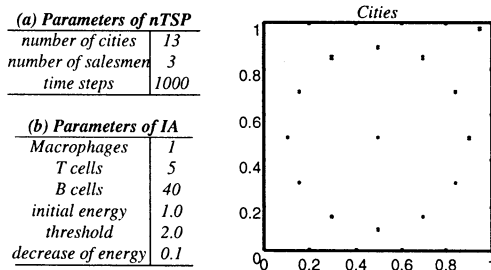


Figure 5: Definition of problem

### 3.3 Results and discussions

In the figure 6, *Cost* is the evaluation of the tour and *Complete* is the degree that satisfies the condition of the solution as n-TSP ( $0 < value \leq 1$ , if the value is 1, the tour composed all cities and no overlap). Though the production of solutions is made by random, the decrease of the cost by communication used MHC (equation 1). However, the algorithm could not find the optimum solution yet. As a reason, it is considered that the environment changes another one before better behavior is made, because importance is

attached to the behavior which memories were used. Therefore, this algorithm may be necessary to shift of the search scope from local optimum because the decrease of the cost isn't seen after 500-time step.

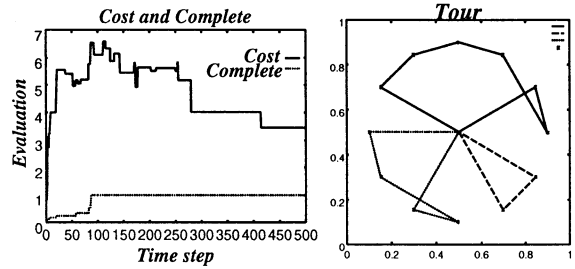


Figure 6: Cost and tour in the last step

## 4 Conclusion

In this paper, we proposed the immune algorithm for multi-agent system, composed immune network for pre-adaptation and post-adaptation, and MHC for mutual cooperation between agents. This algorithm was applied to simple n-TSP, and validity was examined. As future works, this algorithm becomes clear some points about its behaviors.

### Acknowledgements

We would like to appreciate Prof. Y. Itoh (Lymphocyte Biology, Section Laboratory of Immunology) with stimulating discussion. This was supported in part by the Ministry of Education of Japan under Grant 10780240 in Aid for Scientific Research.

## References

- [1] Thomas Back (editor): "Proceedings of The Seventh International Conference on Genetic Algorithms, Morgan Kaufmann", 1997.
- [2] Charles A. Janeway, Jr / Paul Travers / Simon Hunt / Mark Walport : "Immunobiology : The Immune System in Health And Disease", 1997.
- [3] Y. Ishida, H. Hirayama, H. Fujita, A. Ishiguro, K. Mori.: "Immunity-Based Systems and Its Applications", CORONA, 1998
- [4] D.E. Goldberg,: "Genetic algorithm, search optimization and machine learning", Addison Wesley, 1989.
- [5] K. Mori, M. Tsukiyama, T. Fukuda.: "Application of an immune algorithm to multi-optimization problems", T. IEE Japan, Vol.117-C, No.5, pp.593-598, 1997.
- [6] S. Forrest, A.A. Perelson.: "Genetic algorithm and the Immune system", Proc. of 1st Workshop on PPSN, pp.320-325, 1990.

## Application of Competitive Co-evolution Algorithm to Iterated Prisoner's Dilemma

Moeko Nerome  
moe@eva.ie.u-ryukyu.ac.jp  
Graduate School of Science and Engineering  
University of the Ryukyus  
1 Senbaru Nishihara, Okinawa  
903-0213, JAPAN

Satoshi Endo, Koji Yamada, Hayao Miyagi  
{endo, koji, miyagi}@ie.u-ryukyu.ac.jp  
Faculty of Engineering  
University of the Ryukyus  
1 Senbaru Nishihara, Okinawa  
903-0213, JAPAN

### Abstract

Competitive co-evolution is well known as one of adaptive algorithms. In some optimization problem, this algorithm can acquire the solutions of the problem effectively and has the ability to search a global space [1, 2]. To further analyze the behavior of this algorithm, we apply it to the Iterated Prisoner's Dilemma (IPD) [3, 4]. First, we design the competitive co-evolution model for application to the IPD. Next, this model is implemented to acquire strategies, and then computer simulations are carried out. Using the computer simulations, we analyze the relative abilities of the evolved strategies.

### 1 Introduction

Competitive co-evolution algorithm is a well known as adaptive algorithm such as neural networks and genetic algorithms. Some of the attractive features of this algorithm are its ability to acquire the solutions of a problem adaptively, searching for better solutions and having the capability to search a global space. In our previous works, we adapted this algorithm as an acquisition strategy in a board game, Tic-Tac-Toe game, and succeeded in the acquisition of strategies [2]. In this paper, we apply this algorithm to the Iterated Prisoner's Dilemma (IPD) [3, 4] to further analyze its behavior. The difficulties in the IPD are that each player must predict the other player's action and the strength of a player varies for every opponent. Thus, the approach of acquiring strategies adaptively is necessary for the IPD. We design the competitive co-evolution model for applying this algorithm to the IPD that contains complexity. This model is implemented to acquire strategies, and then computer simulations are designed. From the computer simulations, we observe the behavior of this algorithm by analyzing how to evolve strategies.

### 2 Iterated Prisoner's Dilemma (IPD)

The Prisoner's Dilemma is a two-person non-zerosum game, which has been used for both experimental and theoretical investigations of cooperative behavior. In this game, each player chooses his action from two alternatives, Cooperate or Defect. Each player is assigned numerical values for each pair of choices. An example of such a payoff matrix for the players (player1:P1, player2:P2) is shown in Table 1.

Table 1: The payoff matrix of the Prisoner's Dilemma

$P_1 / P_2$	Cooperate (C)	Defect (D)
Cooperate (C)	$R:3 / R:3$	$S:0 / T:5$
Defect (D)	$T:5 / S:0$	$P:1 / P:1$

In this game, each profit Reward (R), Temptation (T), Sucker (S) and Punishment (P) must satisfy the following relations.

$$T > R > P > S \quad (1)$$

$$2R > T + S \quad (2)$$

In these relations, no matter what the other does, the selfish choice of defection yields a higher payoff than cooperation. Also if both defect, both do worse than if both had cooperated.

If the game is viewed as a single event, each player finds defection to be the optimal behavior, regardless of the opponents action. However, in the Iterated Prisoner's Dilemma (IPD) in which two players will meet again, the most optimal choice of action is difficult. In the IPD, each player must have each strategy which decides its action based on some rules.

The difficulties of IPD are as follows:

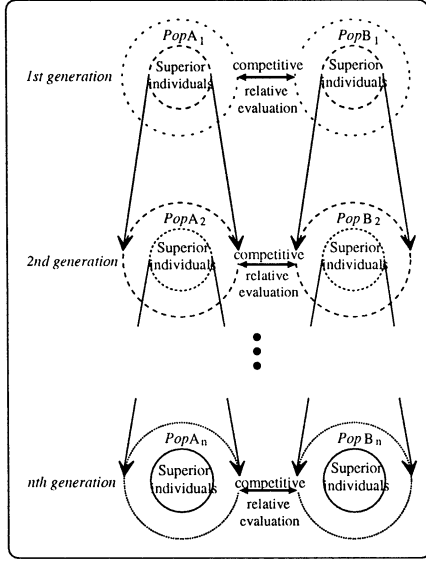


Figure 1: Competitive co-evolution

1. Each strategy must decide its action by predicting the opponent's next action.
2. The strength of each strategy varies for every participant.

Therefore, it is necessary for the IPD to model the strategy that action is decided based on the finite history and the approach of acquiring strategy in a dynamically changing environment.

### 3 Competitive co-evolution algorithm

#### 3.1 Competitive co-evolution

Competitive co-evolution refers to the simultaneous competitive evolution of two or more populations. In competitive co-evolution, the fitness of an individual in one population is based on direct competition with some individuals from another population. We represent the competitive co-evolution as shown in the Figure 1. Such interactions have been well known as a phenomenon which occurs in nature and modeled by various researchers [1].

#### 3.2 Competitive co-evolution algorithm

Figure 2 shows the competitive co-evolution algorithm used in this study.

**step1:** Creating two populations,  $P_a$  and  $P_b$ .

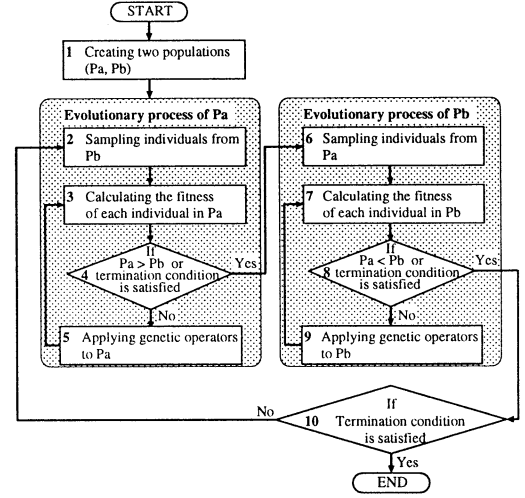


Figure 2: Competitive co-evolution algorithm

- step2:** Sampling individuals from  $P_b$  as a set of evaluating  $P_a$ .
- step3:** Calculating the fitness of each individual in  $P_a$  by competing with a sampling set from  $P_b$ .
- step4:** If  $P_a$  predominates over  $P_b$  or termination condition is satisfied, go to step6, otherwise go to step5.
- step5:** Applying genetic operators to  $P_a$  using the fitness calculated in step3. And return to step3.
- step6:** Sampling individuals from  $P_a$  as a set of evaluating  $P_b$ .
- step7:** Calculating the fitness of each individual in  $P_b$  by competing with a sampling set from  $P_a$ .
- step8:** If  $P_b$  predominates over  $P_a$  or termination condition is satisfied, go to step10, otherwise go to step9.
- step9:** Applying genetic operators to  $P_b$  using the fitness calculated in step7. And return to step7.
- step10:** If the termination criterion of this algorithm is not satisfied, return to step2.

In this algorithm, from step2 to step10 is called one competitive co-evolution generation.

As mentioned above, the competitive co-evolution algorithm acquires better individuals by keeping on predominating over each other.

We show the characteristics of this algorithm which has a mechanism of the interactive improvement of individuals as follows:

1. In the problem where a fitness function giving absolute evaluation can't be set, it is possible to acquire the solutions of the problem effectively using this algorithm. This is because each individual is given relative evaluation by competing with their co-evolving individuals.
2. By searching the solutions adaptively the algorithm obtains better solutions.
3. This algorithm has the ability to search a global space, that is global search algorithm.

## 4 Competitive co-evolution model for the IPD

We analyze the behavior of the competitive co-evolution algorithm in the IPD which contains complexity. Thus, we design the competitive co-evolution model for the IPD.

### 4.1 Definition of populations

In the competitive co-evolution model, there are two populations. For the IPD, one population ( $P_a$ ) is a set of the evolving strategies, the other ( $P_b$ ) is a set of their opponents. The fitness of each individual in  $P_a$  is based on direct competition with some individuals from  $P_b$  (in this paper, all individuals).

When  $P_a$  improves more than  $P_b$ , the part of  $P_a$  and  $P_b$  is replaced.  $P_b$  is a set of the evolving strategies,  $P_a$  is a set of their opponents. In this way, the evolution progresses by replacing both parts.

### 4.2 Genetic coding of strategies

We use the genetic coding of strategies that it is devised by Lindgren [4].

In the model, each individual has a finite memory. Each individual decides own action based on the finite history. An  $m$ -length history consists of a series of previous actions. An  $m$ -length history is expressed by a binary number is shown in the following (0 for defection and 1 for cooperation).

$$h_m = (a_{m-1}, \dots, a_1, a_0)_2 \quad (3)$$

where,

- $a_0$  : the opponent's last action
- $a_1$  : the individual's own last action
- $a_2$  : the opponent's next to last action

Since a deterministic strategy of memory  $m$  associates an action to each  $m$ -length history, it can be specified by the following binary sequence.

$$S = [A_0, A_1, \dots, A_{n-1}] \quad (n = 2^m) \quad (4)$$

This sequence then serves as the genetic code for the strategy that chooses action  $A_k$  when history  $k$  turns up.

## 4.3 Evolutionary Dynamics

We use the Genetic Algorithm (GA) as evolutionary dynamics of strategies in the competitive co-evolution algorithm. Therefore, the fitness of strategies and genetic operators are described.

### 4.3.1 Fitness of strategies

The fitness of strategy is calculated based on the following function.

$$F_i = \frac{1}{Opp} \sum_j g_{ij} \quad (5)$$

where,

- $F_i$  : the fitness of  $i$ th strategy
- $Opp$  : the number of opponents
- $g_{ij}$  : the average score for the strategy of  $i$  playing against opponent of  $j$

### 4.3.2 Genetic operators

We use the roulette selection for reproduction of strategies in next generation, one-point crossover for keeping schema and three kinds of mutations for existing various strategies in a population. The mutations are described as follows.

- point mutation : this flips a gene in the chromosome.  
example : [00]  $\rightarrow$  [01]
- gene duplication : this attaches a copy of the chromosome to itself.  
example : [01]  $\rightarrow$  [0101]
- split mutation : this randomly removes the first or second.  
example : [1001]  $\rightarrow$  [01]

The gene duplication and the split mutation causes the length of chromosome to change.

#### 4.4 Simulation results and discussion

To analyze the behavior of competitive co-evolution algorithm, this model is implemented to acquire strategies. The parameters for the simulation are:

Table 2: Parameters for simulation

population size	50
one-point crossover rate	0.2
point mutation rate	0.01
gene duplication rate	0.01
split mutation rate	0.01
termination of applying genetic operators	100
termination of competitive co-evolution	100

The evolution of strategies starting with the memory one strategies [00], [01], [10] and [11] is shown in Figure 3 and Figure 4. The average score of strategies in populations ( $P_a$  and  $P_b$ ) is presented in Figure 3. In the Figure 3, acquired strategies are classified roughly in three kinds (A is defective strategy, B is Tit-for-Tat-like strategy and C is cooperative strategy). From the result, the evolution of B against A, the evolution of C against B, the evolution of A against C, can be confirmed respectively. This fluctuation of strategy's ability is confirmed in the wide width of the vibration in particular. We consider that the reason of the fluctuation is because for the IPD, the best strategy varies for every opponent. Therefore, when this model is applied to the problems such as IPD, the model causes the fluctuation in the evolution of solutions.

On the other hand, the existence percentage of strategy of every memory in population ( $P_a$ ) is presented in Figure 4. Our prediction is that if the strategies of long memory exist in population, the strategies will keep being given the chance of survival. However, actually, strategies of long memory and short exist in the population alternately. We consider that strategies of short memory are easy to converge one ability more than strategies of long memory because the number of short is fewer.

From these results, we observe a relation between the width of the fluctuation of the average score and the existence percentage of strategies of every memory. When the generation causes a wide width, strategies of short memory exist in both populations or either. On the other hand, in the generation of the narrow width, strategies of long memory exist. When the width of fluctuation is narrow, the cooperative strategies exist in populations. Therefore, the strategies of long memory should exist in populations for emerging the cooperative relation between two populations.

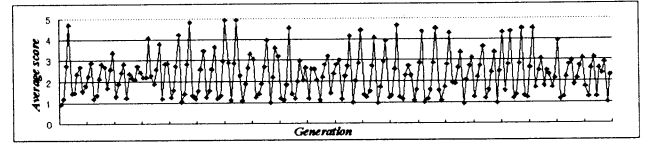


Figure 3: The average score of strategies in populations

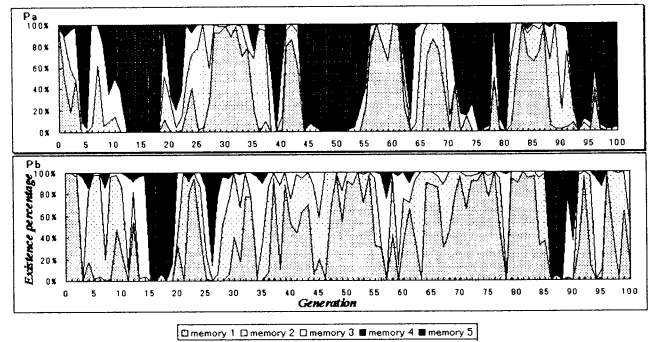


Figure 4: The existence percentage of each strategy in population

#### 5 Conclusion

In this paper, we designed the competitive co-evolution model for application to the IPD. In the computer simulation, we observed that this model caused the fluctuation in the evolution of solutions, when it was applied to the problems such as IPD. In the future work, we need to adjust the parameters for constructing the cooperative relation between strategies and apply it the IPD with noise.

#### Acknowledgements

The authors are grateful to "Support Center for Advanced Telecommunications Technology Research" for their financial support.

#### References

- [1] W.D. Hillis : *Co-evolution parasites improve simulated evolution as an optimization procedure*, Artificial Life II, Addison-Wesley(1991).
- [2] Moeko Nerome, Koji Yamada, Satoshi Endo, Hayao Miyagi : *Competitive Co-evolution Based Game-Strategy Acquisition with the Packaging*, Proceedings of the Second International Proceedings of the KES'98, pp.184-189, 1998.
- [3] R. Axelrod : *The Evolution of Strategies in the Iterated Prisoner's Dilemma*, Genetic Algorithms and Simulated Annealing, Pitman(1987).
- [4] Kristian Lindgren : *Evolutionary Phenomena in Simple Dynamics*, Artificial Life II, pp295-312, 1991.

## Vector Lyapunov functions method in stability and control theories for logic- dynamical systems.

V.M.Matrosov

Stability and Nonlinear Dynamics Research Center  
of Mechanical Engineering Research Institute  
of Russian Academy of Sciences, Moscow, Russia  
e-mail: vladimir@matrosov.mccme.ru

### ABSTRACT

The theory has been developed of logic dynamic systems (LDS), representing complex systems with changing structure, that allows reconfiguration and hybrid description, containing a mathematical models as well as logical equations. The vector Lyapunov functions (VLF) method has been elaborated for their qualitative investigations; algorithmic principle for inference of theorems on dynamical properties of stability and control type has been invented.

### INTRODUCTION. <sup>1</sup>

Complex (large scale) systems, contain of several subsystems long time functioning can be disconnected or connected again to the initial system, which can cause faults, damages or emergency situations. Connectively stable systems must have a requiring stability properties and be able to save them under various structural changes caused by faults of elements, subsystem connection or disconnection, subsystem content changing e.t.s.

VLF-method [1-4] has been developed for analysis of dynamical properties of complex nonlinear system with various structural changes [5-6]. Side by side with system structure changes in several cases the changes in space dimension as well as impulse coordinate changes can be allowed.

The theorems on properties such as stability or boundedness, dissipativity, invariance type under structural perturbations controllability, fault tolerance, survival and stability under constantly acting perturbations, has been stated using comparison principle with VLF. The general approach to obtain conditions of saving properties under various structural changes has been stated.

Structural changes has to be understood

as transformation of system structural matrices  $E$  [5] in some time moments.

In applications such structural changes could be caused by faults, damages, and recovering of subsystems elements, control reconfigurations, functioning mode changes e.t.c. [6].

### 1.VLF-METHOD FOR COMPLEX NONLINEAR SYSTEMS WITH CHANGING STRUCTURE.

If system state vector is supposed to change dimension than the system coordinates has to be defined when it's turned on. State coordinates of connecting subsystems has to be defined through coordinates of other subsystems, and impulse changes of state coordinates also allowed under structural changes. In practice this corresponds to chopping coordinate changes of control system with changing structure, reconfiguration by logical knowledge processing or logical automatas [7].

#### 1.1. Stability for systems with structural changes.

Let suppose that  $i(t)$  piece-constant function of time, with values from discrete set  $L = \{1, \dots, l\}$ . Let's consider a system, which

<sup>1</sup>This work was sponsored by Russian Foundation for Fundamental Researches (RFFI) grant N 99-01-01164



in  $i$ -th structural state can be represented by differential equation

$$\dot{x} = f(t, x, i(t)), \quad x \in R^{n_i(t)}, \quad (1.1)$$

$$i \in L, t \in R_+ \setminus \Theta$$

where  $\Theta = \{t_k : 0 \leq t_0 < t_1 < t_2, \dots, t_k \rightarrow \infty \text{ when } k \rightarrow \infty\}$  - set of structural changes moments, and relation

$$x(t_k, i(t_k) = i) = \varphi_{ij}^k(x(t_k - 0, i(t_k - 0) = j))$$

$$t_k \in \Theta \quad (1.2)$$

in change process from  $j$ -s to  $i$ -th structural state in  $t_k$ , time moments  $x(t_k - 0, j) \in R^{n_j}$ ,  $x(t_k + 0, i) \in R^{n_i}$ ,  $i, j \in L$ . Let's suppose functions  $f$  to be continuous if  $t \in R_+ \setminus \Theta$  and each  $i \in L$ , right continuous if  $t \in \Theta$  and satisfy existence and prolongability conditions for each structural state. In addition let's suppose  $f(t, 0, i) = 0$ ,  $t \in R_+$ ,  $i \in L$ , and a functions  $\varphi_{ij}^k$  to be continuous and  $\varphi_{ij}^k(0) = 0$ ,  $i, j \in L$ ,  $k \geq 1$ .

Let's involve a class  $\mathcal{L}$  of continuous on  $t$ ,  $x$  and locally Lipschitzian on  $x$  in domain  $R_h^{n_i(t)}$  for each  $i \in L$ ) vector functions  $V(t, x, i) : R_+ \times R_h^{n_i(t)} \times L \rightarrow R^{N_i(t)}$ , which satisfy continuity conditions on  $t$  on systems (1.1) (1.2) solutions, except, if needed, in the points of discontinuity of  $x(t)$ , where  $V(t, x(t, i(t)), i(t))$  right continuous.

Let's define also

$$\begin{aligned} D^+V(t, x, i) = \\ \lim_{h \rightarrow +0} \sup \frac{1}{h} [V(t+h, x+hf(t, x, i), i) - V(t, x, i)] \\ \leq g(t, V(t, x, i), i), \quad t \neq t_k \end{aligned} \quad (1.3)$$

For all  $t \geq t_0$ ,  $x(t) = x(t, i(t), t_0, x_0, i_0)$  is any solution of the system (1.1), (1.2), existing when  $t \in R_+$ .

*Comparison theorem.* Let next conditions are hold

1)  $V \in \mathcal{L}$ , satisfied differential inequality (1.3) for  $t \neq t_k$ ,  $(t, x) \in R_+ \times R_h^{n_i}$ , where  $g : I \times R_+^{n_i} \rightarrow R^{n_i}$  for each  $i \in I$ ,  $g(t, 0, i) \equiv 0$  and  $g(t, y, i)$  quasi monotonous nondecreasing function of  $y$ ;

2)  $h_0 > 0$  exists such that  $x_0 \in R_{h_0}^{n_i}$  imply  $\varphi_{ij}^k(x) \in R_h^{n_i}$  and

$$V(t, \varphi_{ij}^k(x, i), i) \leq \psi_{ij}^k(V(t - 0, x, j), j),$$

where  $t = t_k$ ,  $x \in R_h^{n_j}$ ,  $\psi_{ij}^k : R_+^{n_j} \rightarrow R_+^{n_i}$  nondecreasing of  $y$ ;

3) for all  $(x, t) \in R_h^{n_i} \times R_+$  and  $i \in L$

$$b(\|x\|) \leq \|V(t, x, i)\| \leq a(\|x\|),$$

where  $a, b \in \mathcal{K}$  (class of Hahn) and

$$\|V(t, x, i)\| = \sum_{l=1}^{N_i} V_l(t, x, i).$$

Then the stability properties of trivial solutions of comparison system (CS)

$$\frac{dy}{dt} = g(t, y, i), \quad t \neq t_k$$

$y(t_k, i(t_k) = i) = \psi_{ij}^k(y(t_k - 0, i(t_k - 0) = j)$ , imply corresponding stability properties for solution  $x = 0$  of the system (1.1), (1.2).

Stability type dynamical properties for system with structural changes has been investigated. Someone can see the development of this results for analysis of logic-dynamical control systems stability, and also possibilities of its applications for dynamical reliability, vitality and safety of technical systems.

## 1.2 The developing systems of processes with structural changes.

We consider description of mathematical model using concepts of developing system of processes (SP) and abstract control system [1,2], which are quite general axiomatic constructions unifying and comprehend practically any system of models, which has been used for describing a states, control and perturbations, hybrid data and knowledge. Processing of expert and model knowledge and estimations by means of artificial intelligence.

It can be given like logical processes, dynamical knowledge bases in discrete time.

As far as for control processes with perturbations the content and structure of the system, contemporary its state space, could depend on control choice and change with perturbations realization.

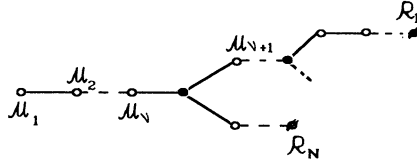
The developing system of processes  $r$  could consist of several interconnected subsystems, described by different classes of differential, difference, functional differential equations, inclusions and other relations [1]. In simplest case discrete continuous nonlinear interconnected control system with structural changes and perturbations is described

$$x_i(t+1) = F_i(t, Ex_i(t), u_i, p_i) +$$

$$+ \tilde{F}_i(t, Ex(t), u, p), \quad t \in N, \\ (i = 1, \dots, k') \quad (1.4)$$

$$\frac{dx_i(t)}{dt} = EX_i(t, x_i(t), u(t_0), p_i) + \\ + E\tilde{X}_i(t, x(t), \tilde{E}u(t), p) \quad t \in T = [0, \tau) \\ (i = k' + 1, \dots, k) \quad x_i(t) \in R^{n_i(t)}, \\ (x_1(t), \dots, x_k(t)) = x(t) \in R^{n(t)}, \quad n = \sum_{i=1}^k n_i(t), \quad \tilde{F}_i, \tilde{X}_i \text{ -interconnections. (with control } u \text{ and perturbations } p)$$

Definitions  $\mathcal{B}$  of dynamical properties (DP) of SP  $r$  proposed to be in form of a graph with typical quantifies and conclusion formulas in it's points, expressed through the vectors  $\rho$  and  $\rho_0$  of estimation functionals and estimation sets  $H_{t_0}^*, U, P^*, H_{t_0}, P(t, \epsilon), P_{t_0}^0(\delta) \dots$



$m_\alpha$ -typical quantifies  $\bullet - \wedge$  or  $\vee$ ,  $\mathcal{R}$ -final formula. For example,  $\rho = (\rho^1, \dots, \rho^l)$ .

$$\mathcal{R} \equiv (x \in P(t, \epsilon))$$

$$\hat{m} \equiv (\forall z)\beta \Rightarrow \dots \quad \check{m} \equiv (\exists z)\beta \wedge \dots$$

$$P(t, \epsilon) = \{x \in X^t : \rho(t, x) \leq \epsilon\}, \quad \epsilon \in R_+^l$$

For example, in [4] there is considered a definition  $\mathcal{B}$  of safety and stabilizability like complicated dynamical property that consist of conjunction of two simple properties.

It means existence of normative control such that for any normative perturbations and initial data each process do not exit from the safety set and for any time interval of normative length  $\Theta^H$  process holds in stability set on each time intervals of sum prolongability less than  $\Theta^s$ .

### 1.3 Comparison principle for theorem derivation

Algorithms for inference of comparison theorems had a following structure

$$(\exists v, s, v_p, w, r_c : \mathcal{M}, C) \mathcal{B}_c \Rightarrow \mathcal{B}$$

where  $v(t, x) = x_c$ ,  $s(t_0, u) = u_c$ ,  $v_p(t_0, x_0, p) = p_c$ ,  $w(t) = t_c$ ,  $r_c$  - CS,  $C$  - some connection formula,  $\mathcal{B}_c$  is corresponding DP for CS  $r_c$ .

It was proved by sequence of heuristic rules, where quantifier formula  $C$  gives some conditions which are derived from the quantifier formula of studied DP definition from which it was obtained by heuristic and logic approach (see [1,2,4]).

Let for given DP  $\mathcal{B}$  by analysis of  $\mathcal{M}, C, \mathcal{B}_c$  conditions  $\mathcal{M}^*, C^*, \mathcal{F}(\mathcal{B}_c)$  do not contain  $x, x_c$  are obtained such that  $\mathcal{M}^*, C^*, \mathcal{F}(\mathcal{B}_c) \Rightarrow \mathcal{M}, C, \mathcal{B}_c$ , or only  $C^* \Rightarrow C$

**Definition.** Vector comparison function  $(v, s, v_p, w)$  for system  $r$  is called vector Lyapunov function (VLF) with respect to dynamical property  $\mathcal{B}$  if it satisfy conditions  $\mathcal{M}^*, C^*, \mathcal{F}(\mathcal{B}_c)$ .

There are obtained *algorithmic principle for logical inference of the theorems on dynamical properties*

$$(\exists(v, s, v_p, w) : \mathcal{M}^*, C^*) \mathcal{F}(\mathcal{B}_c) \Rightarrow \mathcal{B}$$

Lets denote  $\tilde{W}$  -the set of structural influences and suppose that the description of system  $r$  under consideration depend on structure, represented by structural matrices  $S$ . For system  $r(S)$  someone need to obtain conditions of dynamical properties changing under structural changes, defined by set of structural interactions  $\tilde{W}$ .

Vector Lyapunov function for system  $r(S)$  with respect to dynamical property  $\mathcal{B}(S)$  is vector comparison function [6]  $v(S)$  for the system  $r(S)$ , if the formula  $\mathcal{B}(S)$  is true for it.

If for system  $r(S)$  the VLF  $v(S)$  exists with property  $\mathcal{B}_v(S)$ , comparison system  $r_c(S_c)$  with property  $\mathcal{B}_c(S_c)$  and a property  $\mathcal{B}_v(S)$  is invariant with respect to  $\tilde{W}$  and  $\mathcal{B}_c(S_c)$  is invariant with respect to  $\tilde{W}_c$  than the system  $r(S)$  holds investigated dynamical property  $\mathcal{B}(S)$  with respect to definite set of structural influences  $\tilde{W}$  (see [6]).

Applying this statement to comparison theorem it's easy to state whenever the considered dynamical property holds under all structural perturbations, defined by structural influences  $\tilde{W}$ .

If a comparison system has been chosen in form of upper solutions system of finite dimensional differential equation and sufficient conditions for connection formula could be denoted in form of Chaplign differential inequalities than there is effective procedures for transformation comparison theorems into theorems on dynamical properties.

Let VLF  $(v, s, v_p)$ ,

$$v(t, x) = (v^1(t, x), \dots, v^k(t, x)), \quad t \in T, x \in H$$

is introduced,  $t_c \equiv t$ . For control systems described by a differential equations in  $R^{n,(t)}$

$$\frac{dx(t)}{dt} = F(t, x(t), u, p) \quad (1.5)$$

$v$  is like continuous function satisfy next conditions: along any solutions  $x(t, h)$ ,  $h \in H^N$  almost everywhere for  $t \in T(x)$

$$\bar{D}_+ v(t, x(t, h)) \leq F_c(t, v(t, x(t, h)), s(t_0, u), v_p(h)). \quad (1.6)$$

As a CS in this case it's naturally to chose the differential equation

$$\frac{dx_c}{dt} = F_c(t, x_c, u_c, p_c), \quad (1.7)$$

$$x_c \in X_c^t = A(t) \subseteq R^k$$

with right side solution  $x_t$ ,  $F_c$  defined in domain  $A$  satisfying quasimonotonicity on  $x_c$  there, measurable (in Lebesgue sense) and bounded in norm by summable function in each compact. This provide an existence of generalized 2-type solutions [2]  $x_c(t, h_c)$  absolutely continuous.

*Theorem 1.* Let exists VLF ( $v, s, v_p$ ) and CS (1.7) satisfying (1.6) and other above noted conditions, including condition: comparison system has a property  $B_c$ . Then safety and stabilisability is held for control system (1.5).

Analogous theorems could be formulated for development processes with discrete time ( $T = N$ ), and also on general case for discrete - continuous processes, described by (1.3). The last case require some additional investigations, concerned with VLF method development VLF construction technique in application to considered problems.

## 2. VLF- METHOD ORIENTED SOFTWARE.

There are elaborated 3 software packages.

2.1 Software package for inference of theorems of VLF-method.

More than three hundred comparison theorems for the various properties of a system of processes, abstract controlled systems and differential systems were obtained. This is achieved for the first time in the world practice of computer theorem inference [2].

2.2 Software packages for analysis of stability type DP, quantitative estimations and parametric synthesis of controllable systems

with perturbations (see [3]). Various mechanical, technical and automic control theory problems, which can be described by ordinary differential equations with perturbations, controls and nonlinearities may be studied.

2.3. Software System DYNAMICS - a Tool for Computer-Aided Modeling of Fail-safe Control Systems described by continuous-discrete logic-dynamic equations [7].

Since 1992, the SS DYNAMICS has been successfully applied in the leading Russian enterprise engaged in elaboration, creation and exploitation of the fail-safe gyromoment attitude control systems (ACS) of the spacecrafts (SCs).

The SS operates in IBM PC/AT.

## REFERENCES

- [1] V.M.Matrosov, I.Yu.Anapolsky, and S.N.Vassiljev *Comparison method in mathematical system theory*, Nauka, Novosibirsk, 1980, p.453.
- [2] V.M.Matrosov S.N.Vassiljev R.I.Kozlov and others *Algorithms for theorem derivation in vector Lyapunov functions method* Novosibirsk, Nauka, 1981.
- [3] R.Z.Abdullin, V.M.Matrosov and others. *Vector Lyapunov Functions in Stability Theory*. World Federation Publishers Inc. 1996, pp.394.
- [4] V.M.Matrosov "Vector Lyapunov function method: theory and application to complex industrial systems". Proc. Conf. Control of industrial systems. Belfort, France, May, 1997. pp. 137-149.
- [5] D.D.Siljak *Large scale dynamic systems: stability and Structure* N.Y.: North-Holland 1978.
- [6] V.M.Matrosov, A.I.Malikov *Vector Lyapunov functions in analysis of dynamical properties for system with structural changes*. Izvestia RAS, Theory and systems of control. N1, 1998.
- [7] V.M.Matrosov, V.A.Rayevsky, M.F.Reshetnev, Ye.I.Somov, *Nonlinear methods and software for dynamic investigations of fail-safe gyromoment attitude control systems of suttelites*. Proc. of I International conference on nonlinear problems in aviation and aerospace (1996) .Embry-Riddle Aeronautical University, Florida, USA, 1997.

# AHP Coefficients Optimization Technique Based On GA

Takeaki Toma & Mohammad Reza Asharif

University of Ryukyus, Department of Information  
Engineering

E-mail: j94033@ie.u-ryukyu.ac.jp

## 1 Abstract

This study provides information about solution of the problems with Analytic Hierarchy Process(AHP) evaluation function using Genetic Algorithm (GA). Analytic Hierarchy Process is a decision making method which is widely used on Operations Research Decision making. The AHP uses a multiple – criteria, pairwise-weighting method to systematically quantify perceptions generated by primarily qualitative criteria. We propose to combine AHP and GA to seek the optimum solution. We realized this combination of the GA and AHP by inserting the gene codes into the pairwise matrix for the evaluation function. By this combination of two different approaches, we could realize the robust algorithm.

**KeyWords:** Genetic Algorithm, Optimization, Operations Research, Decision Making, Management science

## 2 Introduction

To optimize an evaluation function for solving problems, we need to find a method which gives the related parameter. In conventional approach, for instance, finding the minimum to mean square error (MSE), we use a gradient method. That is making the

derivative of the MSE to zero will give us the minimum point. However, in problem solving such as management of the human science, that we need a fuzzy decision to make the above gradient approach does not work well. We have been studying the possibility of applying AHP technique to iterative problems. The characteristic features of AHP are its robustness and iterative approach to seek the optimum solution. Since AHP is widely used for “human decision making”, if the problem consists of hundreds of factors, it would be almost impossible to operate AHP iteratively by human.

Here, we propose the GA algorithm for operating the AHP iteration for seeking the optimum solution. Instead of human, AHP operated by GA seeks the optimum solution by its heuristic evolution.

## 3 Analytic Hierarchy Process

AHP is a decision making method invented by Dr. Thomas L. Saaty.<sup>1</sup> It is based on the innate human ability to use information and experience to estimate relative magnitudes through paired comparisons. The AHP algorithm consists of 4 major parts. Construct a problem structure hierarchy by decomposing a complex problem in search of cause-effect explanations. Next, evaluate each criteria's significance by comparing same level factors. Finally construct the general evaluation of each solution alternatives, and select the best alternative.

Make a hierarchy as a representation of a complex problem in a multilevel structure whose first level is the goal followed successively by levels of factors, criteria, sub-criteria, and so on down to a bottom level of alternatives. For example, if the problem is “Selecting the best computer”, then we can make a problem structure tree (Figure 1)

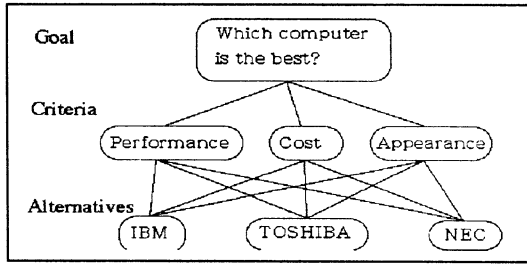


Figure3-1: AHP Hierarchy

To get the weights of each criteria, Satty proposed to use the “Iegenvector” of criteria-comparisoin matrix. Satty proposed to use the Iegenvector as the criteria weights from its criteria comparison matrix. Suppose that we wish to compare a set of n objects in pairs according to their relative weights. Denote the objects by  $A_1, \dots, A_n$  and their weights by  $w_1, \dots, w_n$ . The comparewise is represented by a matrix of underlying ratios(assumed to exist) as follows.

$$\begin{array}{c}
 A1 \ A2 \ A3 \ \dots \ \dots \ \dots \ \dots \ \dots \ \dots \ \dots \ \dots \ \dots \ A_n \\
 \begin{array}{c}
 A1 \left[ \begin{array}{cccc} \frac{w_1}{w_1} & \frac{w_1}{w_2} & \frac{w_1}{w_3} & \dots & \frac{w_1}{w_n} \\ \frac{w_2}{w_1} & \frac{w_2}{w_2} & \frac{w_2}{w_3} & \dots & \frac{w_2}{w_n} \\ \frac{w_3}{w_1} & \frac{w_3}{w_2} & \frac{w_3}{w_3} & \dots & \frac{w_3}{w_n} \\ \dots & \dots & \dots & \dots & \dots \\ \frac{w_n}{w_1} & \frac{w_n}{w_2} & \frac{w_n}{w_3} & \dots & \frac{w_n}{w_n} \end{array} \right] \\
 A2 \\
 A3 \\
 \dots \\
 A_n
 \end{array}
 \end{array}$$

Clarify the ratio of each criterion which represent those judgements with meanigful numbers on a pair wise comparison matrix by evaluating the importance, significance of smaller problem to anothers to Elicit judgments that reflet ideas, feelings or emotions. And, input the number into the pairwise matrix according to the “importance of one criterion” with others. In the paired comparison approach of the AHP, one estimates ratios by using a fundamental scale of absolute numbers. The absolute number from the scale is an approximation to the ratio  $W_i/W_j$ . The derived scale tells us what the  $W_i$  and  $W_j$  are. This is a

central observation about the relative measurement approach of the AHP and the need for the fundamental scale. Thus, instead of assigning two numbers  $W_i/W_j$ , we assign a single number drawn from the fundamental 1-9 sale to represent the ratio  $(W_i/W_j)/1$  as follows.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{bmatrix}$$

Numerial Values	Definition
1	Equally Important or preferred
3	Slightly more imortant or preferred
5	Strongly more important or preferred
7	Very strongly more important or preffered
9	Extremely more important or preferred
2,4,6,8 Reciprocals	Intermediate values to reflect compromise Used to relect dominance of the second alternative as compared with the first

Figure 2

This process is done for every pair .

This matrix has positive entries everywhere and satisfies the reciprocal property  $a_{ji} = 1/a_{ij}$ .

It is called a reciprocal matrix. We note that if we multiply this matrix by the column vector  $(w_1, \dots, w_n)$  we obtain the vector  $nw$ . That is:

$$Aw = \lambda w$$

We get the weight of each criteria from this matrix by obtainig the Iegenvector of this matrix A. Then, we can get the weight vector by obtaining the Iegenvector w. Satty<sup>2</sup>

The final step is a weightening process that uses these priorities synthesize the overall importance of the criteria, subcriteria and alternatives, and the highest ranking alternative is chosen.

Let H be a complete hierarchy with h levels. Let  $B_k$  be the priority matrix of the k th level,  $k = 2, \dots, h$ .

If  $W'$  is the global priority vector of the pth level with respect to some element z in the (p-1)st level, then the priority vector W of the qth level ( $p < q$ ) with respect to z is given by the multilinear form,

$$W = B_q B_{q-1} \dots B_{p+1} W'$$

Thus, the global priority vector of the lowest level with respect to the goal is given by ,

$$W = B_h B_{h-1} \dots B_2 W'$$

\*3

#### 4 AHP coefficients Optimization Technique based on GA

Our algorithm is based on the combination of both GA and AHP.(Figure4.1) Analytic Hierarchy Process has its strength on solving difficult , complicated problems with multi criteria.

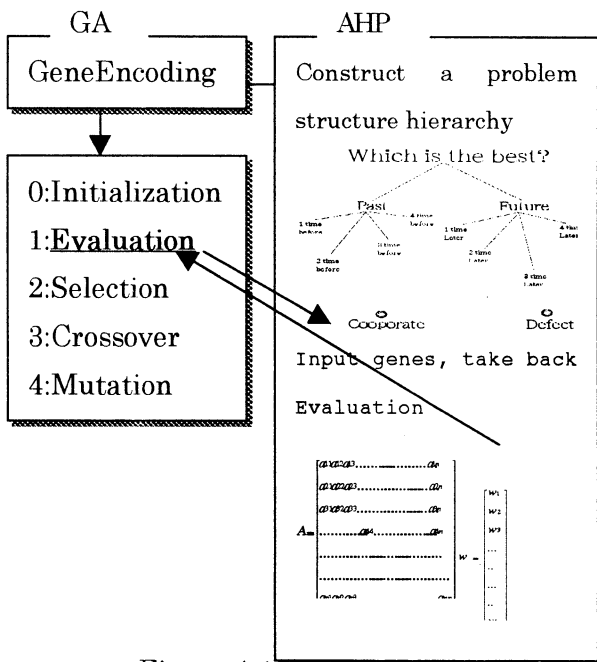


Figure 4-1

##### 4.1 Gene Encoding

The technique for encoding solutions may vary from problem to problem and from genetic algorithm

to genetic algorithm. Here, We use AHP pairwise matrix for encoding. (Figure 4.2). Gtypes are described as interger from 1 to 9 for AHP reciprocal matrix. These gene codes are copied into the AHP pairwise matrix.(figure 4-3)

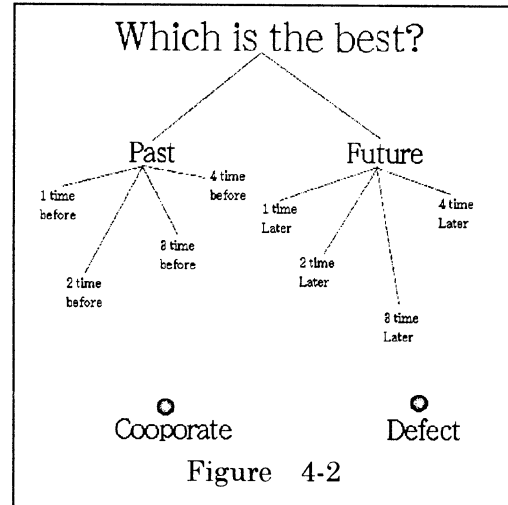


Figure 4-2

##### 4.2 Initialization

The initialization technique generates a population of random integer string. This population makes up the first generation of chromosomes.

##### 4.3 Evaluation Function

Next, each chromosome in the population is evaluated by the AHP evaluation function, and fitness value for each chromosome are computed by the fitness technique.

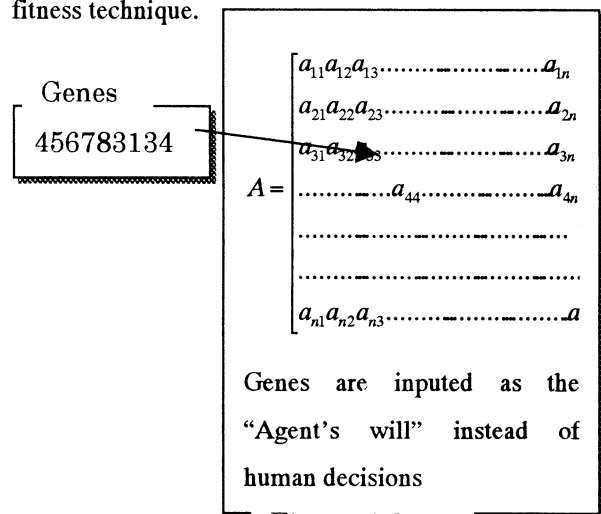


Figure 4-3

The AHP evaluation function is the link between the genetic algorithm and the problem to be solved. The AHP evaluation function takes a chromosome as input and returns the chromosome's performance on the problem to be solved.

#### 4.4 Operator

In reproduction, we use the parent selection technique to pick two parent chromosomes. The reproduction module applies the crossover and the mutate operator to generate two new chromosomes, called children. We introduced the AHP Consistency Index for checking the "consistency level" of gene codes.

$$C.I = \frac{\lambda_{\max} - n}{n - 1}$$

If the C.I exceeds 0.1, it means "No consistency" on the AHP reciprocal matrix <sup>4</sup>, therefore these genes are regarded as lethal genes. The operator may cause those new chromosomes to differ from their parents. After the reproduction, new chromosomes have been produced. These new chromosomes are evaluated, their fitnesses, are calculated, and they replace the current chromosomes to form the next generation. These generational cycles continue until adequate individuals have been produced. At this point, GA halts.

Now GA begins a series of cycles of replacing its current population of chromosomes by a new population. Each of these cycles produce new generation of chromosomes. If all goes well throughout this process of simulated evolution, an initial population of unexceptional chromosomes will improve and parents are replaced by better and better children. If all has gone well, randomly generated population of

chromosomes have evolved so that later generations contain individuals that are better than any found earlier. The best individual in the final population produced can be a highly-evolved solution to the problem.

## 5 Conclusion

It was impossible to solve huge-scale multi-criteria problems. Our method "AHP coefficients Optimization Technique based on GA" can resolve this type of problems by using GA Heuristic iterative approach. Our approach has shown good result on solving some Optimization problems like TSP or Prisoner's Dilemma and other Optimization problems.

## Reference

1. Thomas L. S(1994), "Fundamentals of decision making and priority theory"
2. Thomas L. S(1994), "Fundamentals of decision making and priority theory" : pp 77-79
3. Thomas L. S(1994), "Fundamentals of decision making and priority theory" : pp 112
4. Thomas L. S(1994), "Fundamentals of decision making and priority theory" : pp 80-84
5. Thomas L. S (1982), "The logic of priorities"
6. Golden B.L, E.A.Wasil, P.T.Harker(Eds.), The Analytic Hierarchy Process Applications and Studies.
7. Holland, J.H.(1975) : Adaptation in natural and artificial systems, University of Michigan Press.
8. Rechenberg, I.(1986) : Evolution strategy and human decision making, Human decision making and manual control, Willmeyer, H.P.(ed.) , North-Holland.

# Study on Cooperative Agents through Adaptive Focal Point

Sadamu Yamauchi   Koji Yamada   Satoshi Endo   Hayao Miyagi  
Faculty of Engineering  
University of the Ryukyus  
Nishihara, Okinawa, Japan

## Abstract

In this paper, we discuss the cooperative action of multiagents, using the environment-adaptive type function creation. The focal point is well-known as a method of sharing information without explicit communication between agents. There is the cleanup room problem as one of the problem setting of cooperative action. We consider task handling by the focal point algorithm for this problem. Furthermore, we consider the creation method to fit a change of environment.

## 1 Introduction

A sharing of information among agents is an important problem for realizing cooperative action of robots in multiagents environment.

Cooperative action considered here is a problem in which the ability of several agents is focused and handled as one task. For example, consider the case "there is a thing which cannot be lifted by one person, and cooperation of other persons is necessary in order to lift." A task in this case is "lift a load" and cooperative action is "cooperation of other persons in order to lift a load". In this example, it is important whether joint ownership of information with someone else is involved or not, for example in "It is impossible for me to lift this load alone (I need help from somebody)" and "Is it not impossible for that person to lift that load? (Does he need help?)". There is cleanup room problem in one of the problems by several setting of cooperative action. [1][3]

We consider task handling by focal point algorithm of sharing information only by observation and not communication in this problem. And we consider the creation method of functions to fit a change of environment and uncertainty.

## 2 Cooperative action of robots

### 2.1 Cleanup Room Problem

We deal with cleanup room problem as problem setting of cooperative action. This problem is put on how many objects that should be finished in environment, and this will order the robots to push the objects. Characteristic feature of this problem is that the ability of one robot to process an object is established, and whether it becomes necessary for cooperative action with other robots. In other words, cooperative action becomes necessary for the accomplishment of task.

$$\sum_i (\text{Power of robot } i) \geq (\text{Weight of Target}) \quad (1)$$

Therefore, each robot needs to aim at equal object in order to accomplish a task.

### 2.2 Focal Point Algorithm

Focal Point Algorithm is well-known as a mechanism of making agreement for a long time. For example, the point that can agree between things fully even if we do not communicate specifically, like 1/2 at distributing a thing between two persons. The place which is distinguished from others extremely conspicuously is called focal point.

We show an example which represents focal point by a function in the following case.

An attribute value about object  $x \in X$  or perception information about  $x$  through a sensor is given by,

$$F(x) = (f_1(x), f_2(x), \dots) \quad (2)$$

However,  $f(x)$  seems to be  $f_{color}, f_{size}$ , and there is a case of continuation value in case of disintegration. Agents begin to select one object among those inside set  $X$  of an object using this attribute  $F$ . But we add probability  $P(x)$  that object  $x$  is observed to an element deciding FP because the probability that object



$x$  is observed is different depending on environment state of arrangements of an object. Because focal point is decided by the next function value using these elements, we have.

$$G_i(F_i(x))_{xp_i} \quad (3)$$

We use focal point search, that is a search method by focal point algorithm, here in order to let robots do cooperative action.

### 2.3 Example of Cleanup Room Problem

We next show an example which uses focal point algorithm for Cleanup Room Problem. Focal point algorithm is used with a part doing progress direction decision of a robot here.

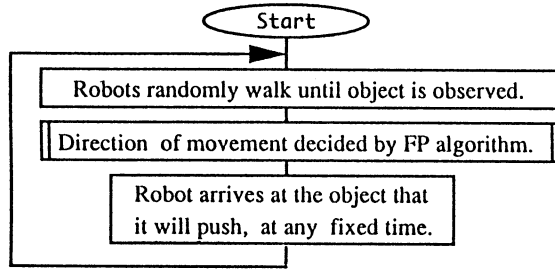


Figure 1: Movement of robot

Let the robots work by the control as in Figure 1. The robot continues by random walk till it observes a target object, and then the selected target object is decided by focal point algorithm when objects are observed. A robot decided with target object in order does not need to compare with the case when only one object is observed, and it is then decided the object progress direction.

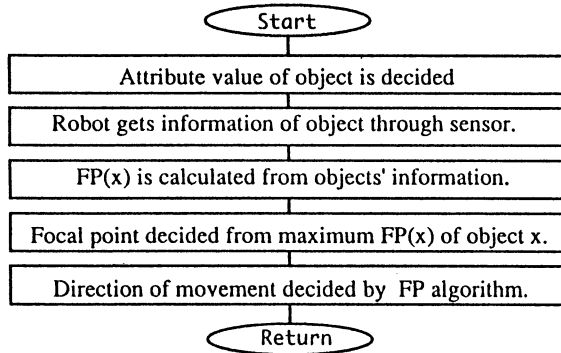


Figure 2: Focal Point Algorithms

Figure 2 is an example to decide progress course using focal point function in cleanup room problem. In this

example, calculate focal point function by an attribute value of object, and through this it is decided when a robot has several observed objects which way becomes progress course.

Focal point function is established as follows.

- Order size of object as a function.  
(Collect robots with a big object or a small one)
- In the case that processed an object, establish give reward value. And enumerate object by size of reward value. (Collect robots with a high object of reward value)

### 2.4 Focal Point Search

When letting cooperative action, there is need to share information between the agents. In this place we use focal point search to share information only by observation information. We define focal point search used here as following,

- Focal Point Search  
Order it by adding priority sequence to objects. By this robots decide progress course to a direction of high object of priority sequence when there are several observed objects.

## 3 Environment-Adaptive Focal Point Function

### 3.1 Focal Point Function

In focal point function  $FP(i)$  in order to bring in focal point search, agents calculate  $FP(i)$  for all objects  $i$ , and decide priority sequence.

Fenster, Kraus, Rosenschein used focal point function represented by (4).

$$FP(x) = \sum_{all f_i} R(f_i(x)) + 0.5E(f_i(x)) \quad (4)$$

However, suppose attribute set  $v_x$  of all objects of observation being done to be,

$$\begin{aligned} \text{Rareness: } R(v) &= 1/(\{v_k | v_k = v\}) \\ \text{Extremeness: } E(v) &= (\max(|\{v_k | v_k \leq v\}|, |\{v_k | v_k \geq v\}|)) / (|\{v_k | all k\}|) \end{aligned}$$

This equation is defined as the decision algorithm, even if individual difference fits each element that decides the focal point, that seems to have high agreement by utilizing Rareness and Extremeness. An attribute value of object that each agent catches needs to be in order to keep complete agreement when this

function is used. And a problem of not letting the focal point agree when characteristic of object cannot be distinguished happens. (when for example, there is no element which is an unusual element and a value or extreme is taken).

### 3.2 The Creation of Focal Point Function

It is natural each agent attribute value of an object observed is different when general environment is considered. Accordingly we consider an attribute value of an object that each robot recognizes as a different thing. And we assume that each agent forms a focal point function originally which can cope with it. In this report, we suggest a method to let it agree by changing focal point function itself in the equation(2). In other words,

$$FP(i) = a ft(i) + b fs(i) + c fc(i) + x_i \quad (5)$$

$a, b, c$  : coefficient

$ft, fs, fc$  : Attribute value of each type, size, color.

We use the method that will let it change whether it is not done or whether we would have each agent bring equation(5) beforehand, and the object those values of  $a, b, c$  selected as agreed with a partner. By this method, agents can keep agreement for a difference of value of each agent or attribute value caught. It is regarded that it can cope when it seems not to satisfy the undistinguishable characteristic in the equation(4).

## 4 Experimental Evaluation

Many pieces exist, and the object that should be finished to two dimensions of space supposes the problem that two robots finish those objects. For this problem, the robot decides progress course by a calculation. We do a simulation of two robots finishing objects using calculation results.

### 4.1 Condition setting and Comparison Object

It is assumed that an attribute value of an object observed by a robot is each wrong. When progress course of two robots agrees, it should be the thing that the two robots or an equal object are going to. We suppose it to finish the object that agreed with two robots in order to simplify it here. And compare the speed of task handling with the time that is needed to finish all objects with end condition with a method of 3, function change system 1, function change system 2, random system.

### Attribute value of Object

The object that the robot should finish has a kind of attribute value as in table 1 and the robot is based on this attribute value and decides progress direction using focal point function.

Table 1 Attribute value of Object (left:R1,right:R2)

R1	type	size	color	R2	type	size	color
obj1	1	1	1	obj1	1	1	1
obj2	2	2	2	obj2	2	2	1
obj3	2	1	1	obj3	2	1	1
obj4	2	2	3	obj4	2	2	1
obj5	2	3	3	obj5	2	3	1

### Function Change System 1

- Two robots can see every object in environment as an initial condition and a value of function  $FP(x)$  is big and assumes that conformity degree is high and lifts priority sequence and it is decided which object robots goes to.
- A value of function  $FP(x)$  is big, and it is assumed that conformity degree is high and lifts priority sequence and it is decided which object robot goes to.

example

$$\begin{aligned}
 FP(i) &= ft(i) - fs(i) + fc(i) + x_i \\
 (a &= 1, b = 1, c = 1, x_i = 0) \\
 FP(1) &= ft(1) - fs(1) + fc(1) + x_i = 1 - 1 + 1 = 1 \\
 FP(2) &= ft(2) - fs(2) + fc(2) + x_i = 2 - 2 + 2 = 2 \\
 FP(3) &= ft(3) - fs(3) + fc(3) + x_i = 2 - 1 + 1 = 0 \\
 FP(4) &= ft(4) - fs(4) + fc(4) + x_i = 2 - 2 + 3 = 3 \\
 FP(5) &= ft(5) - fs(5) + fc(5) + x_i = 2 - 3 + 3 = 2
 \end{aligned}$$

Robot decides progress course in obj4

### Function Change System 2

- Calculate function  $FP(x)$  and average of the value. Near to this average of value, lift priority sequence.

example

$$\begin{aligned}
 FP(i) &= ft(i) - fs(i) + fc(i) + x_i \\
 (a &= 1, b = 1, c = 1, x_i = 0) \\
 FP(1) &= ft(1) - fs(1) + fc(1) + x_i = 1 + 1 + 1 = 3 \\
 FP(2) &= ft(2) - fs(2) + fc(2) + x_i = 2 + 2 + 2 = 6 \\
 FP(3) &= ft(3) - fs(3) + fc(3) + x_i = 2 + 1 + 1 = 4 \\
 FP(4) &= ft(4) - fs(4) + fc(4) + x_i = 2 + 2 + 3 = 7 \\
 FP(5) &= ft(5) - fs(5) + fc(5) + x_i = 2 + 3 + 3 = 8
 \end{aligned}$$

Average from  $FP(1)$  to  $FP(5)$  5.6

$$obj1 : |5.6 - 3| = 2.6$$

$$obj2 : |5.6 - 6| = 0.4$$

$$obj3 : |5.6 - 4| = 1.6$$

$$obj4 : |5.6 - 7| = 1.4$$

$$obj5 : |5.6 - 8| = 2.4$$

Robot decides progress course in obj2

### Change method of coefficient

In function change system 1,2, two robots each has original function  $FP(i)$ , and will change coefficient  $a, b, c$  as follows.

1. An initial value of  $a, b, c$  is 0.
2. Add 1 to either of  $a, b, c$ .
3. Calculate function  $FP(x)$  and value of each object.
4. Decide priority sequence of object and select one object.
5. The object that is selected by other agents...
  - When agrees, no change of coefficient.
  - When it does not agree, add 1 to either among coefficients and subtract 2 from the coefficient which had earlier been increased

### Random Search System

- Two robots select one object among five objects at random, and decide on progress direction.

## 4.2 Results

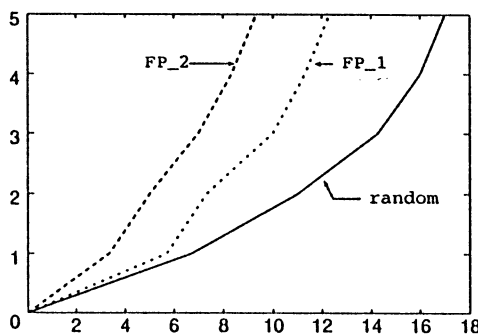


Figure 3 Mean comparison of each search system

In figure 3, X-axe is number of searched objects, Y-axe is number of objects finished. Two kinds of FP function change system show that fast task handling was done compared with a random search than this figure.

## 4.3 Consideration

From the experimental results, it was shown the effectiveness of focal point function change system in task handling. Through focal point function change system, it is possible to achieve a focal point fitting a given change of environment by simply changing a function. It is also found to be more effective in problems of focal point search for elements of an object even when characteristics do not give much information.

## 5 Conclusion

In this report, we showed that it was effective to change focal point function willingly in cooperative action of an agent. Focal point function change system makes it possible to achieve improvement, and sought for creation method of better  $FP(i)$ .

## References

- [1] Matsubara H, Hiraki K, Motomura Y (1994), "An experimental Approach to Learning of Multi-Agent Systems"(in Japanese), Electrotechnical Laboratory,MACC'94.
- [2] M.Fenster, S.Kraus, J.S.Rosenschein (1995), "Coordination without Communication: Experimental Validation of Focal Point Techniques", ICMAS-95.
- [3] Motomura Y, Asoh H, Kuniyoshi Y, Hara I, Akaho A, Matsubara H, Hiraki K (1995), "Formation of the Focal Point in an environment for Multi Autonomous Robots"(in Japanese), Electrotechnical Laboratory,MACC'95.

# On Schemes for Analysis of Stability and Asymptotical Estimations in Critical Cases of Stability Theory.

I.V.Matrossov

*Mechanical Mathematical Faculty  
Moscow State University.*

## ABSTRACT

For ordinary differential equation with holomorphic right side, as a result of comparison of classical [1-4] and modern [5-8] critical cases investigations results in stability theory there is formulated the common scheme for stability analysis.

The algorithmic rule for numerical construction of asymptotical estimations for perturbed motion behavior in critical cases of stability theory has been created. This rule allows to reduce the problem of estimating the solutions to the analogous problem for less dimensional critical subsystem. The Lyapunov case of two zero roots is considered in the details, the complete solution is obtained for this case. In addition the order of perturbation that do not affect stability and estimations has been found [9].

The obtained rules can serve as a base for construction of database and system for investigation of stability and construction of asymptotical estimations for solutions of differential equations on the base of the results known at the verge of the century.

## Introduction.

Let the system of autonomous differential equations in critical case

$$\begin{cases} \frac{dx}{dt} = Ax + X(x, y) \\ \frac{dy}{dt} = By + Y(x, y) \end{cases} \quad (1)$$

is considered, where  $x = (x_1, \dots, x_k)$ ,  $A - k \times k$  - matrix with equals to zero eigenvalue real parts;  $y = (y_1, \dots, y_n)$ ,  $B - n \times n$  matrix with negative eigenvalue real parts;  $X, Y$ - analytical functions of  $x, y$ , that contain only members more than first order in there sums  $(O((x, y)^2))$ .

It's known by A.M.Lyapunov [1] that general form of autonomous system of differential equation with holomorphic right side in any critical case (that can not be analyzed using only linear components) can be reduced to the form (1) by nonsingular linear transformation.

To analyze the stability and asymptotic behavior of the system (1) the first problem is to reduce the investigation of that system

to less dimensional *critical subsystem*

$$\frac{dx}{dt} = Ax + \tilde{X}(x)$$

where  $\tilde{X}(x)$  is a analytical function that can be constructed using  $X(x, y)$  and  $Y(x, y)$ .

The base for this reduction was given by V.A.Pliss [5] theorem on existence of invariant manifold: There exist a smooth invariant manifold  $y = f(x)$ , and any solution of the system (1) converges some solution of the system

$$\frac{dx}{dt} = Ax + X(x, f(x)), \quad y = f(x). \quad (2)$$

exponentially.

## 1.Rule for reduction of initial system analysis to one of critical subsystem.

Let assume that for any given analytical  $X(x) = O(|x|^2)$  function for the system  $\frac{dx}{dt} = Ax + X(x)$  there exist a rule to found the order  $N+1$  of permissible perturbations, that do not affect to a singular point 0 type and asymptotic behavior.

Then use the rule:

1. Transform a matrix  $B$  to Jordan form by nonsingular linear transformation
2. For  $Y(x, 0)$  find a constant  $m_1$  such that  $Y(x, 0) = O(x^{m_1})$ . Assume  $m = m_1$
3. For system  $\frac{dx}{dt} = Ax + X(x, 0)$  find a order of permissible perturbations  $n+1$  using the assumption
4. If  $m < N$ , or  $m_1 \neq m$  (first pass). In system (1) change variables with  $y = \tilde{y} + u^{(m)}$  where  $u^{(m)}$  can be founded as a solution of the equation

$$Bu^{(m)} - \sum_{i=1}^k \frac{\partial u^{(m)}}{\partial x_i} A_i x = -Y(x, 0) + O(x^{m+1}).$$

It's guaranteed that  $\tilde{Y}(x, 0) = O(x^{m_2})$ , where  $m_2 \geq m + 1$ . Initial system changes with transformed one doing following  $X(x, \tilde{y}) = X(x, \tilde{y} + u^{(m)}(x))$ ,  $Y(x, \tilde{y}) = \tilde{Y}(x, \tilde{y})$ ,  $y = \tilde{y}$ ,  $m = m_2$ . Go to step 3.

5. If  $m \geq N$ , than the stability properties and asymptotic estimations for the system (1) and transformed system critical subsystem  $\frac{dx}{dt} = Ax + X(x, 0)$  are equal. The algorithm comes to step 5 in a finite number of operations if the system (2) has any finite order of permissible perturbations.

Proof for stability for  $k = 2$ , see G.V.Kamenkov [3] and for asymptotic behavior for  $k > 0$  see [9].

This theorem shows that the key point of the reduction is to estimate the order of permissible perturbations. There is no general rule to do this in any case but for  $k \leq 2$  the complete solution is obtained:

**Theorem** If the system (1) with  $k \leq 2$  is asymptotically stable then the system (2) has finite order of permissible perturbations.

Proof follows from the estimations given in [9].

Theorem holds true only in the case of analytical  $X, Y$  (for example, the asymptotic stability for the equation  $\frac{dx}{dt} = -\text{sign}(x)e^{-1/x^2}$  can be destroyed by perturbation of any order.

**Remark.** The Rule also gives the estimation for  $y$  through  $x$ :

$$y = u^{(m_1)}(x) + \dots + u^{(m)}(x) + O(x^{m+1})$$

This estimation, if needed, could be make more precise by increasing  $m$  using step 4 (see [9]).

To conclude the part I want to pay attention that the problem has been completely reduced to investigation of purely critical subsystem but not with given  $X$  but with

any possible one. There is proved by Arnold [8] that there is no general algorithm to determine stability for any given system, but this not means that no rule could be constructed but there is no estimation for the number of steps (it can depend upon the system given). The next part of the article most of the positive results concerning such rules will be presented.

## 2. Rules for investigation of critical subsystems in some cases

There are some different criterions to classify the critical cases, one comes from Lyapunov is to use the dimension  $k$  of critical subsystem, the second one used by L.G.Hazin and E.E.Shnol [6] is to determine the codimension  $d$  of the set of some critical case systems in the whole space of the systems (number of conditions that has to be required to obtain the case). Bigger codimension, more seldom the case could appears. Unfortunately the high codimension can not even lower the probability of the case, for example the set of Hamilton systems has infinity codimension.

It's showed in [6] that the complete solution of the problem in general can be obtained for  $d \leq 3$ , in practice in this case the dimension  $n \leq 2$  or the system can be reduced to that case. So here lets consider the case  $n \leq 2$  regardless of codimension, this means that any such case with any finite  $d > 0$  can be analyzed.

### The schemes to analyze the case

$k \leq 2$ .

This case has to be truly considered as one with most detailed investigation.

1. The simplest case is  $k = 1$ . The investigation has been done by A.M.Lyapunov [1]. The system represented in form

$$\frac{dx}{dt} = gx^m + O(x^{m+1})$$

If  $m$ -odd,  $g < 0$  -asymptotically stable,  $x \sim (1 - m)gt)^{\frac{1}{1-m}}$ ,  $t \rightarrow \infty$  otherwise unstable. Order of perturbations  $m + 1$ .

2. In the case of two zero roots with one group of solutions (stability analysis by Lyapunov [2], asymptotical estimations in [9]) the system can be presented in the form

$$\frac{dx_1}{dt} = x_2 + X_1(x_1, x_2) \quad \frac{dx_2}{dt} = X_2(x_1, x_2) \quad (3)$$

where  $X_1(x_1, x_2), X_2(x_1, x_2)$  analytical functions  $X_i = O(|x|^2)$ .

It becomes clear that there is no possibility to determine non asymptotic stability by finite number of arithmetic operations with the members of Taylor series, but us for asymptotic stability and estimation for that case the complete solution can be presented as the following rule:

1. If  $X_1(x_1, 0) \neq 0$  change the variables by  $x_2 = x_1^1 + F(x_1)$ , where  $F(x_1) + X_1(x_1, F(x_1)) = 0$ .

2. Present  $X_2$  in the form  $X_2 = ax_1^\alpha + bx_1^\beta x_2 + O(x_2^2 + x_1^{\alpha+1})$  where constants  $\alpha, \beta \geq 2$  -are natural numbers.

3. If  $\alpha$  -odd,  $\alpha \geq 2\beta + 1$ ,  $\beta$ -even,  $b < 0$ ,  $a < 0$  then:

3.1 If  $\alpha > 2\beta + 1$ , then the solution is asymptotically stable and

$$|x_1| \sim \left( \frac{a(\beta - \alpha + 1)}{b} t \right)^{\frac{1}{\beta - \alpha + 1}}$$

$$x_2 \sim -\frac{a}{b} x_1^{\alpha - \beta}$$

or

$$|x_1| \sim \left( -\frac{b\beta}{\beta + 1} t \right)^{-\frac{1}{\beta}}$$

$$x_2 \sim \frac{b}{\beta + 1} x_1^{\beta + 1}$$

when  $t \rightarrow \infty$ .

3.2 if  $\alpha = 2\beta + 1$ ,  $d = b^2 + 4a(\beta + 1) \geq 0$ . than the solution is asymptotically stable and the estimations are:

$$|x_1| \sim \left( \frac{(b \pm \sqrt{d})(\beta - \alpha + 1)}{2(\beta + 1)} t \right)^{\frac{1}{\beta - \alpha + 1}},$$

$$|x_2| \sim -\frac{(b \pm \sqrt{d})}{2(\beta + 1)} \left( \frac{(b \pm \sqrt{d})(\beta - \alpha + 1)}{2(\beta + 1)} t \right)^{\frac{\alpha - \beta}{\beta - \alpha + 1}}$$

where  $t \rightarrow \infty$ .

4. If  $\alpha$ -odd,  $2\beta \geq \alpha - 1$ ,  $a < 0$ . then without losing generality  $a = -1$ ,  $\alpha = 2n - 1$  (see [2]). Change variables by the rule  $x_1 = rCs\nu$ ,  $x_2 = -r^n Sn\nu$  where functions  $Sn$  and  $Cs$  can be detected as a solutions of the system

$$\frac{dCs(\nu)}{d\nu} = -Sn(\nu) \quad \frac{dSn(\nu)}{d\nu} = Cs^{2n-1}(\nu)$$

with initial conditions  $Cs(0) = 1$ ,  $Sn(0) = 0$ .

4.1 if  $2\beta > \alpha - 1$  after this transformation the system (3) appears in form

$$\left\{ \begin{aligned} \frac{dr}{dt} &= r^{n+1} R(r, \nu) \\ \frac{d\nu}{dt} &= r^{n-1} + r^n \Theta(r, \nu) \end{aligned} \right. \quad (4)$$

Eliminate  $t$  from equations (4) to obtain

$$\frac{dr}{d\nu} = \frac{r^2 R}{1 + r\Theta} = R_2 r^2 + R_3 r^3 + \dots$$

where  $R_2, R_3, \dots$  periodical functions of  $\nu$ .

Find the first members of the solution representation  $r = c + u_2 c^2 + \dots + u_m c^m + \dots$  until nonperiodic function in form  $u_m = g\nu_v$  where  $v$  periodic appears. If  $g < 0$  the solution is asymptotically stable and

$$r \sim \left( \frac{m-1}{m+n-2} t \right)^{-\frac{1}{m+n-2}}$$

For  $x$  and  $y$  the following upper estimations are obtained

$$|x_1| \leq r(t) \quad |x_2| \leq \frac{r^n(t)}{n^{0.5}}$$

4.2. If  $\alpha = 2\beta + 1$  and  $b^2 + 4a(\beta - 1) < 0$  the system appears as

$$\frac{dr}{dt} = br^n Sn^2 \nu Cs^{n-1} \nu + r^{n+1} R$$

$$\frac{d\nu}{dt} = (1 + bSn\nu Cs^n \nu) r^{n-1} + r^n \Theta$$

Exclude  $t$ :

$$\begin{aligned} \frac{dr}{d\nu} &= \frac{brSn^2 \nu Cs^{n-1} \nu + r^2 R}{(1 + bSn\nu Cs^n \nu) + r\Theta} = \\ &= R_1 r + R_2 r^2 + \dots \end{aligned} \quad (4')$$

$R_1, R_2, \dots$  periodical functions of  $\nu$ .

4.2.1. If  $n$  -odd,  $b < 0$ , then the system is asymptotically stable and

$$r \leq e^{-I} \left( -\frac{I(n-1)}{\omega} \left( 1 - \frac{|b|}{2n^{0.5}} - 2\epsilon \right) \right)$$

$$e^{-I(n-1)t} + \frac{I(n-1)}{\omega} C \right)^{\frac{1}{1-n}}.$$

$$r \geq e^I \left( -\frac{I(n-1)}{\omega} \left( 1 + \frac{|b|}{2n^{0.5}} + 3\epsilon \right) \right)$$

$$e^{I(n-1)t} + \frac{I(n-1)}{\omega} c \right)^{\frac{1}{1-n}}.$$

where  $I = \int_0^\omega \frac{bSn^2 \nu Cs^{n-1} \nu}{1 + bSn\nu Cs^n \nu} d\nu$ . The difference between the upper and lower estimations here shows that the function  $r$  behave when  $t \rightarrow \infty$  like a function  $\frac{1}{t} + \frac{\sin t}{2t}$ .

4.2.2 if  $n$ -even, than use the step 4.1 where in stead of (4) use (4').

5. Otherwise the system is either unstable or not asymptotically stable.

In this case the order of permissible perturbation is determined in [9].

If the initial system is asymptotically stable the process in 4.1 stops in a finite number of steps.

The case of two purely imagine roots can be resolved using the same rule but with  $n = 1$ .

To complete the two-dimension case it is reasonable to consider the case of two zero roots with unique group of solutions. The stability analysis for that case was first considered by I.G.Malkin [4].

The considered schemes could seems to be easy for software implementation, but some problems arises here. The first one appears as the result of computer truncation error, the error makes it impossible to check the equalities such as  $a = 0$ . In the works N.I.Matrossova [10] the interval algorithms has been applied to go around this. The advantage of such approach is that the provable solution can be obtained in comparably simple critical cases of  $k = 1$  or one pair of purely imagine roots.

Now lets turn to multidimensional case. In this case no general scheme exists, but only some rules for the special cases. Here lets describe one interesting scheme for non-resonant case of  $k$  pairs of purely imagine eigenvalues.

Let's consider the critical subsystem in complex form

$$\frac{dz_j}{dt} = i\omega_j z_j + z_j \sum_k A_{jk} |z_k| + O(\|z\|^3)$$

in terms of variables  $\rho_j = |z_j|^2$  together with this system lets consider the simplified system

$$\frac{d\rho_j}{dt} = \rho_j \sum_k a_{jk} \rho_k \quad (5)$$

where  $a_{jk} = \text{Re}A_{jk}$  Next theorem, proved by Molchanov (see [7]) gives the base for the investigation of the system:

The solution  $\rho = 0$  of the system (5) is asymptotically stable if it have not got the invariant growing and neutral rays.

As invariant growing ray lets denote the solution in the form  $\rho_j = \gamma_j t$ .

As neutral ray lets denote any ray that consist of a equilibrium points.

The scheme for stability analysis in this case is the following

1. Shoos the set of indexes  $\{j_1, \dots, j_r\} = I$
2. If the system  $\sum_k a_{jk} \gamma_k = 1, j, k \in I$  has a solution with  $\gamma_j > 0$  than there exist a invariant growing ray, implying the Instability.
3. If the system  $\sum_k a_{jk} \gamma_k = 0, j, k \in I$  has a solution with  $\gamma_j > 0$  than there exist

a invariant neutral ray, stability can not be guaranteed.

4. If all possible sets  $I$  passed the tests 2,3 then the system is asymptotically stable, if not go to step 1.

For the proof refer L.G.Hazin and E.E.Shnol [6].

As it comes clear most of the rules outlined herein are difficult to apply manually, due to great number of steps, but they can serve as a base for construction of computer software for analytical investigation of stability and construction of asymptotic estimations for solutions of differential equations.

## REFERENCES

- [1] Lyapunov A.M. "General problem on stability of motion", Complete works, vol.2, Moscow, Academy of Sciences (AS) of USSR Publ, 1956.
- [2] Lyapunov A.M "Investigation of the one of critical case o the problem of stability of motion". Complete works, vol.2, Moscow, AS USSR Publ, 1956 pp 272-331.
- [3] Kamenkov G.V. "Selected works" v. II, "Science", Moscow, 1972.
- [4] Malkin I.G. "Theory of stability of motions", Moscow, Science. 1966.
- [5] Pliss V.A. "Reduction principal in stability theory", Izvestia of AS USSR Ser.Mathematic v.28, 1964.
- [6]. L.G.Hazin, E.E.Shnol. "Stability in critical equilibrium points", Pushino, 1985.
- [7]. V.V.Roumiantsev in collection of the papers "Mechanics in USSR for 50 years", vol 1, Mosocw, Science, 1970.
- [8] V.I.Arnold "On nonalgebraicity of problem of stability" Functional analysis 1970 v.4 N.3.
- [9]. I.V.Matrossov. Estimations of solutions in critical cases of stability theory". Differential equations (in appear)
- [10] N.I.Matrossova "On the theory of critical cases on investigations of stability of motion", IMASC Annals on computing and applied mathematics, vol. 8., J.C.Baltzer AG Publ., switzerland, 1990.

# Parallel Distributed Architectures of Biologically Inspired Parameter-free Genetic Algorithm for Simulating Ecosystems

Hidefumi Sawai and Susumu Adachi  
Kansai Advanced Research Center  
Communications Research Laboratory  
Kobe, 651-2401 Japan

Sachio Kizu  
Toshiba R & D Center  
Kawasaki, 210 Japan

## Abstract

We describe a novel Genetic Algorithm which we call a Parameter-free Genetic Algorithm (PfGA). The PfGA is inspired by the idea of a biological evolution hypothesis, i.e., the *disparity theory of evolution*. Two different types of parallel distributed architectures of the PfGA are investigated using benchmark problems in the the first International Contest on Evolutionary Optimization at ICEC'96 for simulating artificial ecosystems; One is a uniformly distributed type and another is a *master-slave* type where a specific sub-population supervise immigration of individuals among sub-populations. We compare and discuss the performance of two parallel distributed architectures using parallel processors to get an insight into the nature of ecosystems.

**Keywords:** genetic algorithm, evolutionary computation, parallel distributed processing, ecosystem

## 1 Introduction

The Genetic Algorithm(GA) is an evolutionary computation paradigm inspired by biological heredity and evolution[1]. GAs have been successfully applied to many practical applications such as functional optimization problems, combinatorial optimization problems, and optimal design of parameters in machines [2]. However, some problems for GA have not yet been resolved. One such problem is that the design of genetic parameters in a GA has to be determined by trial and error, making optimization by GA *ad hoc*.

To relieve the user of this kind of adaptive parameter-setting problem, we proposed a Parameter-free Genetic Algorithm (PfGA)[8] inspired by the *disparity theory of evolution*, where no control parameters for genetic operations need to be set as constants in advance. This disparity theory of evolution is proposed by Furusawa et al.[3] based on different mutation rates

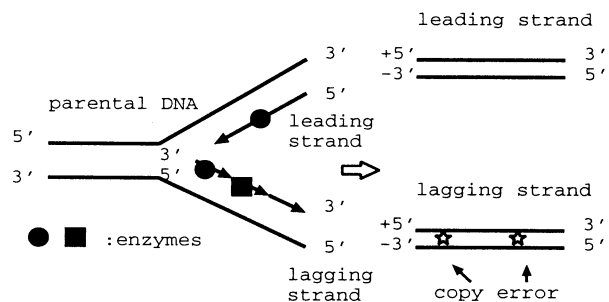


Figure 1: A hypothesis in the *disparity theory of evolution*

in double-stranded DNA.

Parallel and distributed processing of GA has been extensively studied[4] [5], *granularity* ranging from fine- to coarse-grained, and the mode of processing covering both synchronous and asynchronous processing. In this paper we discuss two types of parallel distributed architectures for the coarse-grained model. One such architecture is a uniformly distributed type where some sub-populations asynchronously evolve on multiple processors with some chance of migration among sub-populations. Another architecture is a *master-slave* type where a *master* sub-population supervise other *slave* sub-populations by inspecting the migration of the best individual among all sub-populations. We performed some comparative experiments for the benchmark problems at ICEO (Int. Contest on Evolutionary Optimization)'96 [6] using these two architectures, some interesting and encouraging results of which will be shown.

## 2 PfGA Inspired by Disparity Theory of Evolution

As Charles Darwin claimed in the *Origin of Species*



in 1859[1], a major factor contributing to evolution is mutation, which can be caused by spontaneous misreading of bases during DNA synthesis. Semiconservative replication of double-stranded DNA is an asymmetric process where there is a leading and a lagging strand. Furusawa et al. proposed a *disparity theory of evolution*[3] based on a difference in frequency of strand-specific base misreading between the leading and lagging DNA strands (i.e., disparity model). Fig.1 shows a hypothesis in the disparity theory of evolution. In the figure, the leading strand is copied smoothly, whereas in the lagging strand a copy error can occur because plural enzymes are necessary to produce its copy. This disparity or asymmetry in producing each strand occurs because of the different mutation rates in the leading and lagging strands. Thus maintains *diversity* of DNAs in population as generations proceed. The disparity model guarantees that the mutation rate of some leading strands is zero or very small. When circumstances change, though the original wild type can not survive, selected mutants might adapt under the new circumstances as a new wild type.

The PfGA is inspired by the *disparity theory of evolution*, described above. The population of the PfGA is considered as a whole set  $S$  of individuals which corresponds to all possible solutions. From this whole set  $S$ , a subset  $S'$  is introduced. All genetic operations such as selection, crossover, and mutation are conducted for  $S'$ , thus evolving the sub-population  $S'$ . From the sub-population  $S'$ , we introduce a *family*  $S''$  which contains two parents and two children generated from the two parents. The detailed description on the PfGA is served elsewhere[8].

### 3 Parallel Distributed Processing of PfGA

Generally speaking, parallel processing aims at accelerating the speed of processing. In the case of GA, it aims at reaching better solutions faster than sequential processing by extending the search space. We use a coarse-grained asynchronous model. The coarse-grained parallel GA assigns a sub-population as a unit of processing, and some few individuals are migrated among sub-populations at an appropriate rate. This model is also called an *island model*, and one island (sub-population) constitutes of one *deme* which is a minimum recombinational unit of biological species.

Fig. 2 shows an architecture of uniformly distributed parallel PfGA. One *deme* in this architecture is based on the single PfGA architecture[8]. The whole

population  $S$  is located at the center from which  $M$  sub-populations  $\{S'_i\}$ , ( $i = 1, \dots, M$ ) are extracted. Each *family* shown as  $S''_i$  is extracted from each sub-population  $S'_i$ . If better individuals produced in each sub-population  $S'_i$ , the individuals are copied and migrated among the sub-populations (the bilateral arrows indicate this situation, however, in fact any migration can happen among any sub-populations). One possible migration method is as follows: if the case 1 or 4 happens in some family, the better individual  $C_1$  that is better than its two parents can be copied to other sub-populations as an emigrant. When other sub-populations receive the immigrant, they have to decide to accept it or not because the number of individuals will increase and lead to an explosion of diversity if any immigrants are unconditionally accepted.

Fig.3 shows another type of architecture for parallel PfGA. This can be called a *master-slave* type architecture. Let  $S'_1$  be the master sub-population, and  $\{S'_i\}$ , ( $i = 2, \dots, M$ ) be the slave sub-populations. The master  $S'_1$  always or at some interval supervise itself as well as the sub-populations  $\{S'_i\}$  identifying the best individual among those. When case 1 or 4 happens in some sub-population  $\{S'_i\}$  compare  $C_1$  with the best-so-far individual supervised by the master sub-population. If the former ( $C_1$ ) is better than the latter, copy and distribute it to all other sub-populations including the master one.

These architectures are being implemented using multiple processors connected with local-area networks (LAN). PVM (Parallel Virtual Machine)[7] software is used to evolve the sub-populations parallelly and asynchronously.

### 4 Experiment

Experiments were performed on five function optimization (minimization) problems with 5 and 10 dimensions in the first ICEO[6]. These functions have its own upper and lower bounds defined by its range of variables  $x_i$ . The defined variables were encoded using 22 to 24 bits in Gray coding, depending on each function.

In the ICEO, the following three performance indexes were introduced; the Expected Number of Evaluations per Success (ENES), the best value reached (BV) and the Relative Time (RT). These indexes were measured on every problems of the test bed with 5 and 10 dimensional versions. The ENES index represents the mean number of function evaluations needed to reach a certain fitness value, Value To Reach (VTR), given with each problem. The ENES is computed by

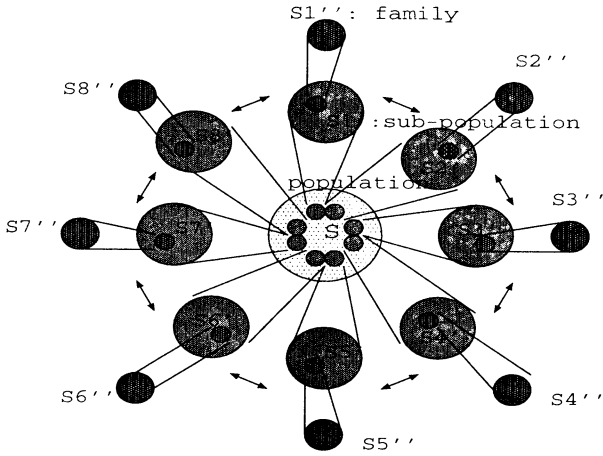


Figure 2: A uniformly distributed parallel architecture of PfGA

running 20 independent runs of the algorithm until the VTR is reached. RT is defined by  $(CT-ET)/ET$ , where CT is the total CPU time to perform the algorithm with 10,000 iterations, and ET the CPU time to perform 10,000 fitness function evaluations. Consequently, the less RT, the more compact the algorithm becomes. Among the nine competitive algorithms including PfGA participated in the ICEO, the PfGA became the second place in 5 dimensional version[8], and the sixth place in 10 dimensional version based on the criterion where the smaller ENES the better.

We compared the performance on two different types of parallel distributed architectures for the PfGA. These two architectures are implemented into a parallel computer (an *alpha machine* by DEC with eight processors) using a network software, PVM. One trial was defined as 10,000 evaluations of individuals in the case of 5 dimensions. An appropriate number of evaluations more than 10,000 (typically 100,000) are adopted for the cases of 10 dimensions because of difficulty in searching their global minima. One hundred trials from different random seeds were performed for each function to get statistics. From the performance results on the function optimization problems, we expect to verify which architecture is better and more appropriate for evolving such ecosystems as the parallel distributed PfGA.

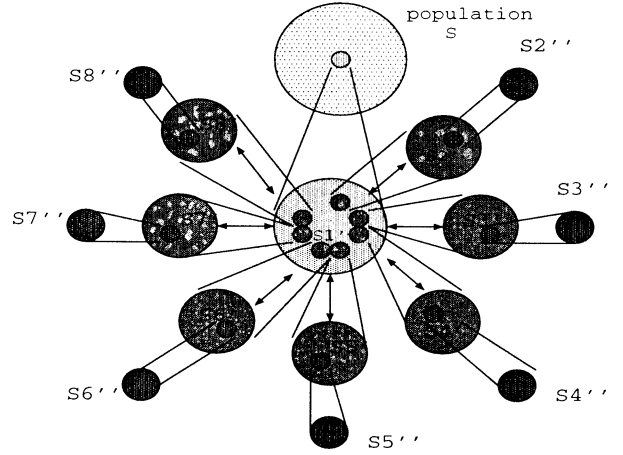


Figure 3: A master-slave type architecture of PfGA

## 5 Experimental Results

Experimental results on two kinds of parallel architectures are shown in Figs.4 and 5 in 10 dimensional version. For 5 dimensional version, the difference between two kinds of architecture was small. In Fig.4, ENES is shown as a function of number of sub-populations with a change from one to 16 (1, 2, 4, 8 and 16). In Fig.5, success rates (%) are shown as a function of number of sub-populations. In these figures, the solid line shows the case of uniformly parallel distributed (PD) architecture, and the dashed line shows the case of master-slave (MS) type of architecture. These results are averaged over five kinds of functions with 100 trials each. We can find that the PD architecture is better than the MS architecture for ENES as the number of sub-populations increases. Also the success rates are almost the same between two architectures, but slightly better for the PD case than the MS case.

## 6 Discussion

We compared two kinds of different parallel architectures using test functions in terms of ability in searching global minima. As a result of that, the uniformly distributed type architecture is found to be better than the master-slave type as an overall performance because the less ENES and more success rate were achieved for the PD architecture. The reason can be thought as follows; The MS type always distributes the best individual to all sub-populations

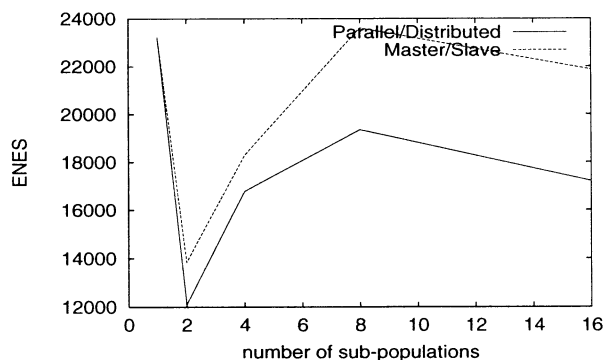


Figure 4: ENES vs. number of sub-populations in 10 dimension

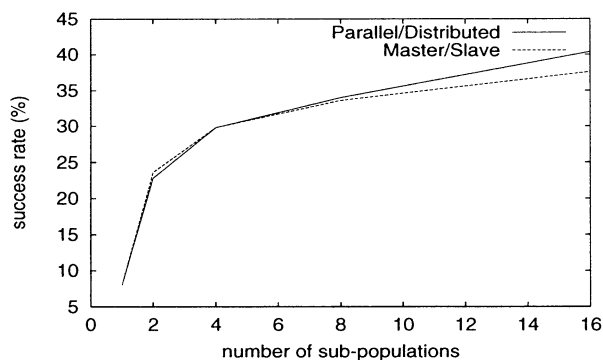


Figure 5: Success rate vs. number of sub-populations in 10 dimension

which may decrease the diversity of ecosystem, whereas the PD type distributes the *locally* best individual to other sub-populations which maintains the diversity of ecosystem. In a huge search space with high dimensions, it is of critical importance to maintain the diversity of ecosystem for finding a global optimum.

## 7 Conclusion

The PfGA is inspired by the *disparity theory of evolution* in which a difference in frequency of strand-specific base misreading between the lagging and leading DNA strands occurs. It shows a rapid evolutionary behavior, does not require the setting of genetic parameters, and is easy to construct. The PfGA is well-suited to parallel distributed process-

ing in which many sub-populations can be evolved either synchronously or asynchronously using parallel processors with some chance of migration between sub-populations. We demonstrated and compared two kinds of asynchronously coarse-grained (island) architectures for the PfGA using multiple processors.

In the current stage, we can simulate an *asexual* ecological system. Furthermore, if we introduce gender (i.e., male and female) into each individual, we can simulate a *sexual* ecological system where sexual reproduction will be realized. These parallel distributed PfGA architectures would also effectively resolve dynamically changing problems and ones in a noisy environment. Therefore, it would possibly be used as a powerful engine for solving such difficult real-world complex problems and further studying ecosystems.

## References

- [1] Charles Darwin, "On the Origin of Species by Means of Natural Selection or the Preservation of Favoured Races in the Struggle for Life," London, John Murray, 1859.
- [2] D.E. Goldberg, "Genetic Algorithm in Search, Optimization, and Machine Learning," Addison Wesley (1989).
- [3] M. Furusawa and H. Doi, "Promotion of Evolution: Disparity in the Frequency of Strand-specific Misreading Between the Lagging and Leading DNA Strands Enhances Disproportionate Accumulation of Mutations," J. theor. Biol., vol. 157, pp 127-133, 1992.
- [4] T.Maruyama, T.Hirose and A.Konagaya, "A Fine-Grained Parallel Genetic Algorithm for Distributed Parallel System," Proceedings of the Fifth International Conference on Genetic Algorithms, pp184-190, Morgan Kaufmann, 1993.
- [5] T.C.Belding, "The Distributed Genetic Algorithm Revisited," Proceedings of the Sixth International Conference on Genetic Algorithms, pp114-121, Morgan Kaufmann, 1995.
- [6] The Organising Committee: H.Bersini et al., "Results of the First International Contest on Evolutionary Optimization (1st ICEO)," 1996 IEEE International Conference on Evolutionary Computation (ICEC'96), pp611-615, 1996.
- [7] Al Geist, Adam Beguelin, Jack Dongarra, Weicheng Jiang, Robert Manchek and Vaidy Sunderam, "PVM UER'S GUIDE AND REFERENCE MANUAL," 1993.5.
- [8] H.Sawai and S.Kizu, "Parameter-free Genetic Algorithm Inspired by 'Disparity Theory of Evolution,'" Int. Conf. on Parallel Problem Solving from Nature (PPSN)-V, Amsterdam, pp702-711, Sep. 1998.

# A Distributed System Inspired From the Immune System: An Application to Control

Yoshiteru ISHIDA

Toyohashi University of Technology  
Toyohashi, Aichi 441-8580 Japan  
ishida@tutkie.tut.ac.jp

## Abstract

Based on the filtering mechanism of the immune system (i.e. positive selection and negative selection), we have proposed an immune algorithm. The algorithm filters agents that are randomly produced; and then activated based on the affinity. By implementing the enhancing mechanism of the resolution of activated agents, and deleting the ones that are not activated, the set of agents become more accurate and efficient in neutralizing disturbances. Several simulations reveal the similarity between the artificial immune system applied to control and the real immune system.

## 1 Introduction

*Neural Networks*, which hinted from neural systems in brain, has been used in many pattern recognition. In recent years, computational method based on the analogy of biological information processing has been paid much attention. *Genetic Algorithm*, which mimic the adaptation of biological systems in genetic level, has been collecting wide attention as a new search method in many fields. Yet another important information processing can be found in immune systems[1, 2, 3, 4, 5, 6, 7], although we cannot be conscious of the activity. The motivation of our research is to (1) explore the principle of information processing done in immune system, and (2) develop information models based on the principle in immune systems, as the neural net is developed based on that in brain systems. Similarly to the immune system, we will consider a multi-agent systems architecture where each agent corresponds to the immune cell. The immune cells are autonomous agents that can recognize and react based on its own decision referring to the information of other agents and environment.

1. Each agent has not only information but a recognizing mechanism itself.

2. Recognizing agent activated by an encounter with the antigen will reproduce its clone to enhance the ability of elimination of the antigen.
3. The reproduction above will be performed with mutation to increase the affinity with the antigen.

We have proposed a distributed algorithm [8, 9] for each agent based on the filtering mechanism of the immune system. The most naive immune algorithm has the following three steps.

1. *Generation of Diversity*: Diversity of recognizer<sup>1</sup> in its specificity is generated.
2. *Establishment of Self-Tolerance*: Recognizers are adjusted to be insensitive to *known pattern*(self) during developmental phase.
3. *Memory of Non-Self*: Recognizers are adjusted to be more sensitive to *unknown pattern* (non-self) during working phase.

In the step of *Memory of Non-Self*, the recognizers which actually recognized the *unknown pattern* during working phase will be activated. Activated recognizers would have any of the following properties resulting in higher affinity with the encountered non-self (for local and cell-based memory as opposed to network memory).

- Elongation of life span or lower death rate to attain immune memory,
- Reproduction of clones (i.e. recognizers of same type),
- Higher rate of mutation.

---

<sup>1</sup>We will use the word "recognizer" for the unit (cell) that has only recognizing and communicating capabilities. The word "agent" is used for the unit that has more intelligence and autonomy; adaptation and self-replication capability as well as recognition capability.

As explored above and in [8, 9], immune algorithm is fundamentally similar to evolution mechanism except it is controlled and accelerated. It is the adaptation process referring not only to environment as in the step of *Memory of Non-Self* but to the self as in the step of *Establishment of Self-Tolerance*. It is an evolution in one individual in one life span rather than in a species in many generations.

The immune algorithm can apply to any system where environment is unpredictable. One typical domain may be fault diagnostic systems where the knowledge of fault is intrinsically unpredictable. In other words, fault is by its nature unexpected, since expectable event can be generally avoidable by design. The technique seems promising, since the diversity of faults is enormous as the system becomes large-scale. Another domain may be control of systems where the knowledge about disturbances imposed is not available. This technique may be especially required for the situation that control signal and disturbance cannot be discriminated in the signal level. We will discuss the application to the control domain in the next section. In conventional control system, disturbance recognizer and controller are physically and conceptually separated. However, in this adaptive disturbance neutralizer, they are identified as recognizers. Recognizers are rather divided by the disturbance patterns upon which recognizers act. To each disturbance pattern of signals, different recognizer which is capable of producing neutralizing signal is supposed to be activated. That is, each recognizer has the capability of recognizing one specific disturbance signal and that of neutralizing the disturbance.

The next section formalizes the adaptive disturbance neutralizer based on the *agent-based* architecture discussed in this section.

## 2 Disturbance Rejection by Immune Algorithm

The transfer functions of plant and controller for numerical simulations are;  $P(s) = \frac{250}{s(s+130)}$ ,  $C(s) = 40$ , respectively. In this specific simulation, the control input  $r(t)$  is assumed to be 0. Thus, the step of *Establishment of Self-Tolerance* is omitted for this simulation. When there are control signals, discrimination between the control signal and disturbance is critical. We further assume that the pair of recognition signal and neutralizing signal is given for those different agents. This pair is actually acquired during long evolutionary process in real biological system.

Disturbance that causes an error shown in Fig. 1 is imposed with period 200 step (1step = .005 sec). Disturbance is assumed to be a periodic disturbance. Fig. 2 shows time evolution of the error. As known from Figs. 2 the adaptive disturbance neutralizer successfully eliminates the disturbance. This means that there is a recognizer that corresponds to the imposed disturbance. The disturbance is also known to be neutralized gracefully from a period to period as step increases. This comes from the adaptation of recognizers, which is the result from adaptation of activated recognizers; higher affinity by higher rate of mutation in this case (as done in the step of *Memory of Non-self* in the immune algorithm).

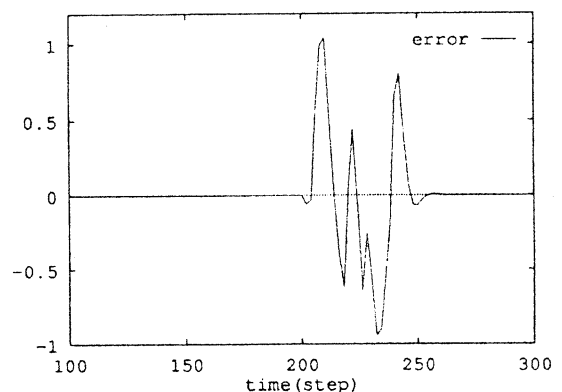


Figure 1: Error caused by the disturbance

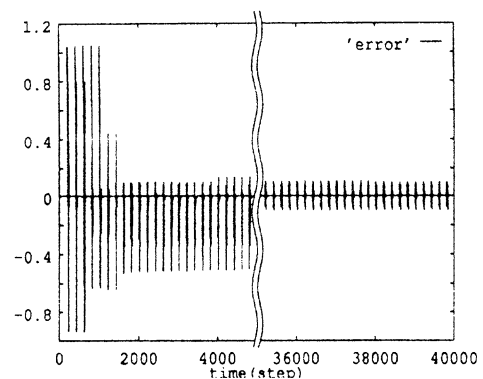


Figure 2: Time evolution of error

### 3 Simulation with Normalized Initial Genes

In this simulation, we use primitive one shown in Fig. 3 as an initial set of genes. In the simulation, ten different genes with different length of base scale are used. Since genes will evolve in the immune algorithm, initial set of gene may be arbitrary. However, diversity is required so that they can compose many type of disturbance signals. Adaptation done in the step *Memory of Non-Self*, gene will mutate. In this simulation, two points of the base of triangle (Fig. 3) changes. Fig. 4 shows an error caused by one period of disturbance. Fig. 5 shows the time evolution of error when the disturbance is imposed with period 100 step (1step = .005 sec). The periodic disturbance is successfully rejected by the system with the normalized initial genes, similarly to the result of the previous section. This means again that there is a recognizer that corresponds to the imposed disturbance.

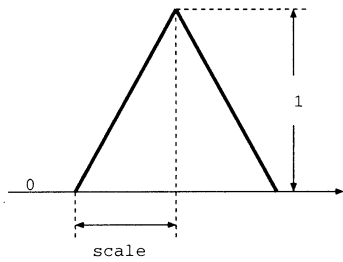


Figure 3: Initial Gene Data; Ten different scales are prepared initially.

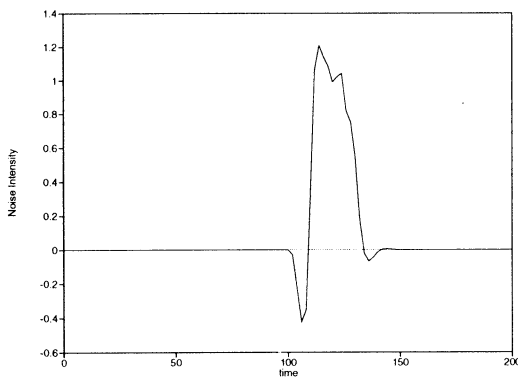


Figure 4: Error caused by the disturbance

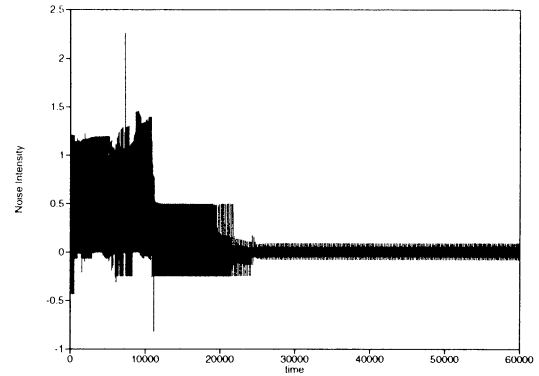


Figure 5: Time evolution of error when the disturbance is imposed with period 100 time step

In another simulation, the responses to the initial (at 0 step) and second (after 40000 step imposition of other disturbances) encounter are compared (Fig. 6). The time evolution of the population of active agents are also monitored as shown in Fig. 7

The disturbance at secondary encounter is more effectively rejected than that at the first encounter. This again comes from the adaptation of the recognizer; elongation of the life time in this case.

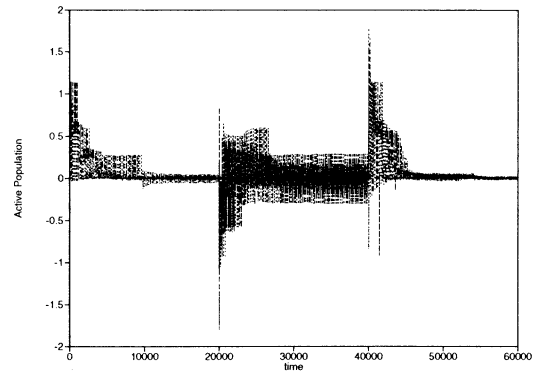


Figure 6: Time evolution of error when first encounter with the disturbance of a type at step 0 and again at step 40000 after imposition of different disturbance from step 20000 to 40000.

It is observed that the population of active agents increases after the new disturbance is imposed and the population becomes saturated after some period. When new disturbance is imposed, the saturated population begins to increase again. Due to this "saturation breaking" phenomenon, there always appears the

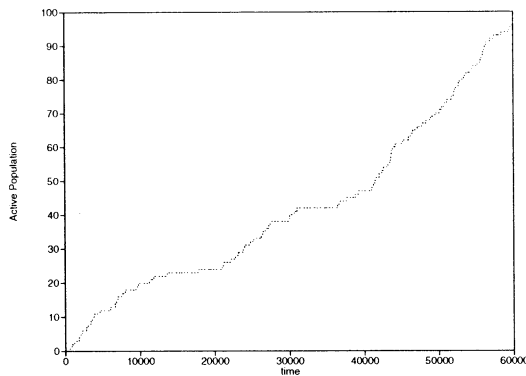


Figure 7: Time evolution of the population of active agents when first encounter with the disturbance of type 1, then type 2 and finally type 1 again.

agent population that can react the control signal (the self) when it is nonzero, even after filtering of step 2 of the immune algorithm. Therefore, similarly to the immune system, the filtering mechanism that continuously delete the self reactive agents (i.e. the agents that can react with the control signal) may be needed.

## 4 Conclusion

By simulations, we have shown that adaptation carried out by the immune system can contribute to the rejection of the periodic disturbances by the immune algorithm which is a filtering algorithm of agents. Phenomena known as “affinity maturation” as well as “immune memory” are also observed in our application of the artificial immune system. However, it is also observed that even after filtering out the agents that can react self (normal control signal in our application), there appears agents that can react with the self. Hence, we have to remove these agents continuously even after the immune algorithm has been applied. This phenomenon is also pointed out in the immune system, hence we have to adopt some counter measure including the one adopted by the immune system.

The immune algorithm proposed may apply to several fields, and seems to be promising for the system where the models of system itself and the environment are not available.

## Acknowledgments

This work has been supported in part by the SCAT (Support Center for Advanced Telecommunications Technology Research) Foundation.

## References

- [1] J. D. Farmer, N. H. Packard, and A. S. Perelson: The immune systems, adaptation, and machine learning; *Physica*, 22D, pp. 187-204, 1986.
- [2] H. Bersisni and F.J. Varela, “The Immune Recruitment Mechanism: A Selective Evolutionary Strategy” *Proc. ICGA 91*, 1991.
- [3] G. M. Edelman: *Bright Air, Brilliant Fire: On the Matter of the Mind*, Basic Books, 1992.
- [4] S. Forrest, A.S. Perelson, L. Allen, and R. Cherukuri: Self-nonsel self discrimination in a computer; *Proc. of 1994 IEEE Symposium on Research in Security and Privacy*, 1994.
- [5] J. O. Kephart: A biologically inspired immune systems for computers; *Artificial Llife IV*, MIT Press, 1994.
- [6] Y. Ishida, “An Immune Network Approach to Sensor-Based Diagnosis by Self-Organization,” *Complex Systems*, Vol. 10, pp. 73-90, 1996, issued in 1997.
- [7] Y. Ishida, “Distributed and Autonomous Sensing based on Immune Network,” *J. of A Life and Robotics*, 1998 to appear.
- [8] Y. Ishida and N. Adachi, “Active Noise Control by an Immune Algorithm” in *Proc. International Conference on Evolutionary Computation (ICEC'96)*, Nagoya, May 20-22, 1996, pp. 150-153.
- [9] Y. Ishida, and N. Adachi, “An Immune Algorithm for Multiagent: Application to Adaptive Noise Neutralization,” in *Proc. IEEE /RSJ International Workshop on Intelligent Robots and Systems (IROS'96)*, Osaka, November 4-8, 1996, pp. 1739-1746.

# **Evolution of Vision System for Stereo Perception by Genetic Algorithm**

Wen Nian Kozo Okazaki  
FUKUI UNIVERSITY

3-9-1 Bunkyo, Fukui 910-8507 Japan

Shinichi Tamura  
OSAKA UNIVERSITY

2-2 Yamadaoka, Suita, Osaka 565-0871 Japan

## **Abstract**

The visual function is easy to visualize its function, and therefore is most easy to understand the brain function among various human activities. We use the genetic algorithm to verify the proposition by a natural manner that it is necessary to have consciousness and abstract consideration ability to carry out higher order functions of the human activities. It may give partly a solution to the question of why the consciousness exists.

In this paper, we propose a binocular stereopsis model to derive the stereo perception ability of the creature.

## **1. Introduction**

Pattern recognition is one of the important tasks in intelligent information processing. The ability of recognizing pattern is the most principal tasks in intelligent system, e.g. a robot. However, ability of conventional pattern recognition system is generally limited. Then it may be valuable to learn from the function of creatures. Here, there is questions whether the vision system of creature is the optimum, or whether another methods can exist. Therefore, to check the necessity of vision system of a creature, we

have been studying the evolution of visual system on a computer assuming that the visual organ of a creature is composed of neural network with arbitrary structure. The structure of the system is determined and evolved by a genetic algorithm.

The visual function is easy to make visualize its function, and therefore most easy to understand the brain function among various human activities. We use the genetic algorithm to verify the proposition by a natural manner that it is necessary to have consciousness and abstract consideration ability to carry out higher order functions of the human activities. It may give partly a solution to the question of why the consciousness exists.

In this paper, we propose a binocular stereopsis model (Fig.1) to derive the stereo perception ability of the creature and further examine the acquisition process of space perception by neural network able to predict future target position which we can regard a spatial reasoning function.

## **2. Binocular stereopsis and 2D Position recognition**

A binocular stereopsis model is shown in Fig.1. The visual depth is obtained by Eq.1.



$$y = \frac{Lf}{X_L - X_R} \quad (1)$$

where  $X_L$  and  $X_R$  denote the positions in left and right eye images corresponding to the point  $P(x,y)$ , respectively,  $L$  is the base line length, and  $f$  is the focal length.

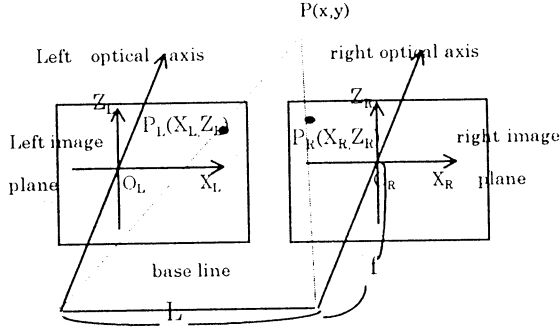


Fig.1 Model of binocular stereopsis

In the 2D plane, the model is changed to one shown by Fig.2. By the Fig.2 we obtain the Eq.2 calculating  $X_L$  and  $X_R$ .

$$\begin{cases} X_L = \frac{x \times f}{f + y} \times KM \\ X_R = \frac{x \times f + y \times L}{f + y} \times KM \end{cases} \quad (2)$$

Where  $KM$  is a function makes all space positions doesn't overlap such that  $X_L, X_R$  are integer. The point  $O$  is set as the origin of  $X_L$  and  $X_R$  to make the value of  $X_L$  and  $X_R$  non-negative number.

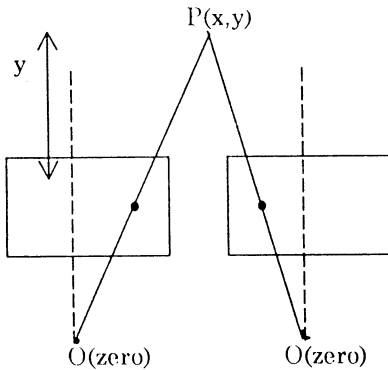


Fig.2 Model of binocular Stereopsis on 2D plane

### 3 Stereo perception and Network

If we start with outputting the distance to the target position, which is proportional to the inverse of the parallax of inputs, it is difficult because of the nonlinearity of the inverse calculation. So we examine the neural network which can output the parallax itself instead of the distance of inputs from a binocular images. Here, the neural network that can output simply the parallax or the distance inversely proportional to the parallax of inputs (only two one-dimensional images corresponding to epipolar lines for simplicity) will be constructed from binocular images. The configuration of the network is shown in Fig.3, where it is composed of two networks independent of each other. The one estimates the distance ( $X_R - X_L$ ), and the other predicts the position ( $X_R + X_L$ ). Inputs are two one-dimensional array expressing  $X_L$  and  $X_R$ . We assume the middle layer is composed of a mutually connected network. Each connection between two units is bidirectional with the same weight. Calculation is done asynchronously. The output (or state) of a unit excited or suppressed is determined by a threshold value. Thus the neural network of the vision system is evolved by genetic algorithm.

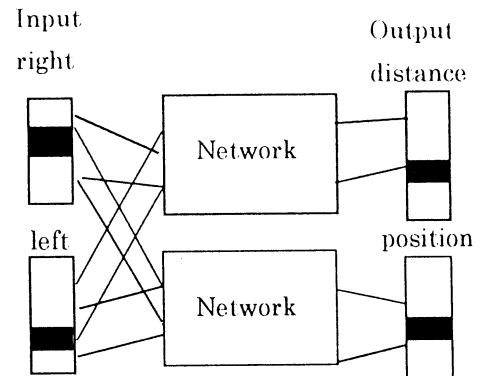


Fig.3 Stereo perception estimation

#### 4 Individual genotype and generation of GA

We must set form of creature's gene when we apply the genetic algorithm. Here, we regard the configuration shown in fig.3 as creature's individual. So we assign the connection weights among all pairs of units and the threshold of process units to the gene code. Thus the gene represents the visual system of virtual creature composed of the neural network

We divide the creatures into two groups to improve survival and evolution: number of creatures in a children's group is 500 and that of an adult group is 250. We adopt the criterion function as shown in Eq.3.

$$p\_o = (DD - ave) \times \frac{ex}{DD} \quad (3)$$

where  $p\_o$  is priority,  $ave$  is average error of hamming distance between two binary numbers of output value and correct value. The  $ex$  is the number of correctly detected objects.  $DD$  is the number of objects estimated.

Here, we set one generation one year and the creature's correct detection value is added 1 for every one generation when the detection is correct. The creatures with ages lower than 11 belong the children's group. In this group, all of the creatures are calculated its criterion, and no operation of death-life and multiplication are applied. When the creature's age is over 10, they are shifted to the adult group and are applied the operation of death-life and multiplication as shown in fig.4.

The number of creatures generated in every generation was set 50. The initial genes are generated randomly from the first to fifteenth generation. In the adult group, the creatures

are arranged in the order of priority (criterion function). The 50 creatures of the next generation are generated from the best 50 creatures by crossing with another 50 creatures in the adult group randomly.

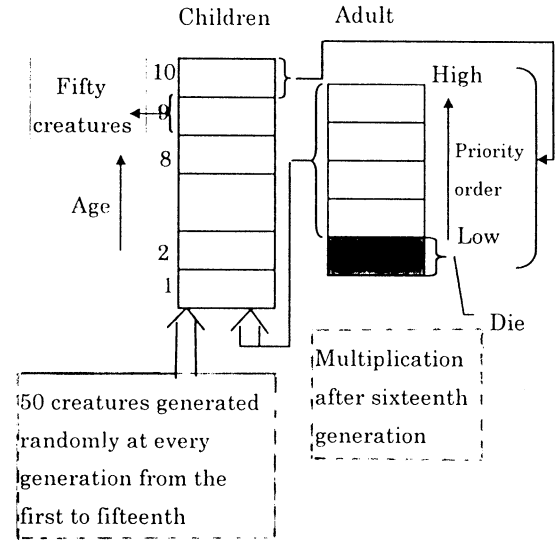


Fig.4 Creature's multiplication

#### 5 Results of experiment and conclusive remarks

Because network1 and network2 is independence of each other, we evolve their by GA independently.

We tested for two space such as 4\*4 and 4\*8. The value of  $y$  is taken from the set {1,2,5,10}. The value of  $x$  is taken from the set {0,1,2,...,n}, where  $n=3$  when it is 4\*4, and  $n=7$  when it is 4\*8. The input value is  $(X_L, X_R)$  corresponding to  $(x,y)$  by Eq.2. Here, the threshold of output units is set as 0. The weight's value between units is set by taking one in the set {-2,-1,0,1,2}. The threshold of process units is set by taking one in the set {0, 1, 2, 3, 4, 5} when the distance is estimated, and set by taking one in the sets {5, 6, 7, 8, 9, 10, 11, 12, 13, 14} when the position is estimated. Table 1 is showing the results

when the number of processing unit is 7,10,15. It is the correct recognition rate of network obtained after 100,000 generations. Here, the crossover was done by uniform rate of 0.70. The mutation rate was taken one from the set {0.15, 0.13, 0.11, 0.07, 0.04} for every 1000 generations cyclically.

In the future step, we will examine the acquisition process of space perception by neural network able to predict future target position which we can regard one of a spatial reasoning function.

Table 1 Correct estimation rate (%)

Space distance*position		Number of process unit		
		7	10	15
4*4	distance	100	100	100
	position	75	81.2	81.2
4*8	distance	100	100	100
	position	47.0	47.0	44.0

## References

- [1] W. Nian, K. Okazaki, and S. Tamura, "Evolution of Vision System by Genetic Algorithm", PROCEEDINGS OF INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS (AROB '97), pp.9-13.
- [2] W. Nian, K. Okazaki, and S. Tamura, "Artificial Visual System Evolution for Stereo Perception by GA", PROCEEDINGS OF INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS (AROB '98), pp.440-443.
- [3] Hopfield, J.J., "Neurons with graded response have collective computational properties like those of two-state neurons", Proceedings of the National Academy of Sciences, 81, pp.3088-3092, 1984.
- [4] Yoh-Han Pao, "Adaptive Pattern Recognition and Neural Networks", Addison Wesley, 1989.
- [5] J.H.Holland, "Adaptation in Natural and Artificial Systems", The Univ. Michigan Press (1975), MIT Press (1992).
- [6] T.Nagao, T.Agui and H.Nagahashi, "Structural evolution of neural networks having arbitrary connections by a genetic method", IEICE Trans. INF. & SYST. Vol. E76-D, No.6, pp.689-697. 1993.

## Immune Algorithm with Adaptive Memory

M. Yonezu

Dept.of Info. & Comp. Sci.  
Keio University  
Kanagawa 223-8522,Japan

T. Yoshida

Dept.of Info. & Comp. Sci.  
Keio University  
Kanagawa 223-8522,Japan

M. Nakanishi,

Dept.of Info. & Comp. Sci.  
Keio University  
Kanagawa 223-8522,Japan

### Abstract

Immune Algorithm(IA) proposed by Mori et al. is one of the computational model of biological immune system for optimization. This algorithm is efficient to keep diversity while searching and so simple as to be applied to any problems which Genetic Algorithms are applicable to. We consider this as extended GA with chaotic wandering, and describe clearly it in this paper.

Although their Immune Algorithm has immune memory as cells, it doesn't have enough ability to utilize the memory for future responses. In this paper, We attach the adaptive memory mechanism to the Immune Algorithm. Our adaptive memory mechanism adopt the adaptive memory search using chaotic neural networks proposed by Nara. Experiments are examined using TSP and knapsack problem to compare it with original IA, and the results of our simulations show that IA with adaptive memory makes a response faster than original IA when solving similar problems repeatedly.

**Keywords** immune algorithm, genetic algorithm, neural network, optimization.

## 1 Introduction

Biological immune system has powerful abilities not only to detect and eliminate antigens, but also to memorize the antigen once met and give a quick response to those memorized antigens. Immune mechanism has not yet been clear enough, however, there are several experimental computational models for immune system. Farmer[4] proposed a computational model of antigen-antibody relation based on the network model of immune systems. There are also researches for engineering purpose. Bersini et al.[1, 2] proposed IRM (Immune Recruitment Mechanism), which is based on the clone selection model of immune system and GIRM (Genetic Immune Recruitment Mechanism) which is IRM with Genetic Algorithm (GA)[5]. The GIRM's

ability to search globally is better than GA, however, it's ability to keep diversity of population is still weak.

Immune Algorithm(IA) proposed by Mori et al.[6] is one of the computational model of biological immune system for optimization. This algorithm is efficient to keep diversity of antibodies[3] while searching and so simple as to be applied to any problems which Genetic Algorithms are applicable to.

Although their Immune Algorithm has immune memory as cells, it doesn't have enough ability to memorize antigen for future response. It is worth improving the utilization of memory cells of IA.

In this paper, we attach an adaptive memory mechanism to the Immune Algorithm using adaptive memory search in chaotic neural networks proposed by Nara et.al[7, 8]. Experiments using TSP and knapsack problem are done to compare it with original IA, and the results of simulations show that IA with adaptive memory makes a response faster than original IA when solving similar problems repeatedly.

## 2 Immune Algorithm

The original Immune Algorithm(IA) proposed by Mori et al.[6] was described in rather complicated way, thus we give a more simple representation here. Our description makes it clearer that the algorithm be so simple as to be applied to any problems which Genetic Algorithms is applicable to. We consider this as extended GA with chaotic wandering.

### Concept

The following correspondences enable us to treat immune system as optimization mechanism in the same way as GA.

- antigen ... given problem
- antibodies ... solutions for the problem
- affinity between antigen and antibody ... fitness

Each antibody is a sequence of alleles. To solve the problem is to find out the best antibody for a given antigen.

### Algorithms

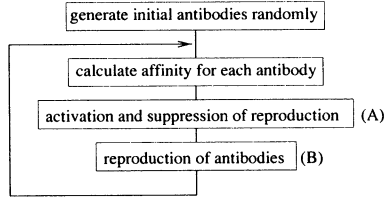


Figure 1: Simple Immune Algorithm.

Architecture of simple Immune Algorithm (Fig.1) is very similar with the simple GA. Affinity between antigen and each antibody is calculated like the calculation of fitness in ordinary GA. On the other hand, Step (A) and (B) are different from GA. Expected value of  $i$  th antibody (which is used for probability of reproduction)  $E_i$  is calculated by:

$$E_i = \frac{A_i}{C_i},$$

where  $A_i$  is the affinity of the  $i$  th antibody and  $C_i$  is the concentration of the  $i$  th antibody, respectively. In reproduction phase (Step (B)), Antibodies whose expected value are low are eliminated and new antibodies are generated and added randomly. Then the genetic operations (crossover and mutation) are done for the set of the antibodies.

General Immune Algorithm(Fig.2) has memory cells to memorize the problem once solved. The memorized cells are used for generating initial antibodies for a new problem (antigen). The number of memory cells is finite, thus a memory management mechanism is indispensable. One of the most simple memory management mechanism is to remove an old memory cell randomly when adding a new antibody to memory space and select antibodies from memory cells randomly when generating initial antibodies.

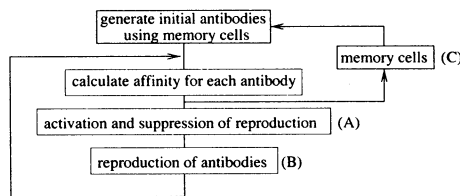


Figure 2: Immune Algorithm with memory cells.

### Diversity of antibodies

The Immune Algorithm is good ability to keep diversity of antibodies. Diversity of antibodies is measured by entropy quantitatively.

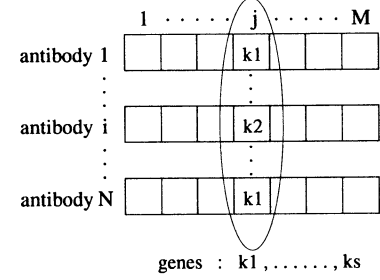


Figure 3: Diversity of antibodies.

The entropy of  $j$  th gene is calculated by:

$$H_j = \sum_{i=1}^S -p_{ij} \log p_{ij},$$

where  $S$  is the number of allele in a gene and  $p_{ij}$  is the provability of the occupance of  $i$  th allele in  $j$  th gene (Fig.3). Then the diversity of whole population  $H$  is calculated by:

$$H = \frac{1}{M} \sum_{j=1}^M H_j,$$

where  $M$  is the number of genes in an antibody.

### Check experiment using 50 cities TSP

Figure 4 shows a check experiment using 50 cities TSP (Traveling Salesman Problem). It shows that simple Immune Algorithm can get better solution than simple Genetic Algorithm (Fig.4(a)) and keep higher diversity of population (Fig.4(b)).

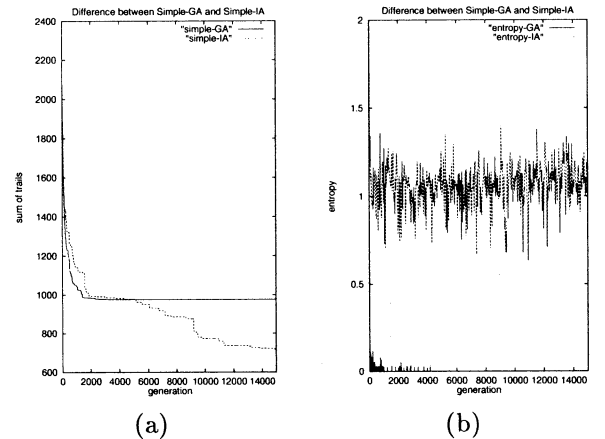


Figure 4: Comparison between Simple IA and Simple GA.

### 3 Immune Algorithm with Adaptive Memory

When generating new initial population, the original immune algorithm selects the antibodies from memory cells randomly. It is considered that there is still room for improvement. In order to utilize memory, we attach a new selecting mechanism based on a chaotic neural network proposed by Nara et al.[7, 8].

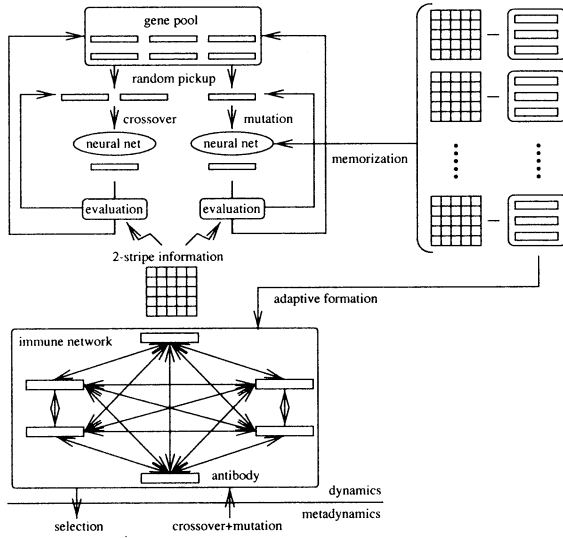


Figure 5: Immune Algorithm with Neural Adaptive Memory.

When selecting antibodies from memory cells, our algorithm (Fig.5) takes the following steps.

1. Construct a mutually connected neural network by embedding memory cells (Fig 6). Weight matrix of the neural network is:

$$W = \sum_{i=0}^{N-1} \mathbf{v}_i \otimes \mathbf{v}_i^+$$

where  $N$  is the number of memory pattern,  $\mathbf{v}_i$  is each memory pattern,  $\mathbf{v}_i^+$  is conjugate of  $\mathbf{v}_i$  and  $\otimes$  is Kronecker product.

2. Represent given problem by pixel pattern, since this algorithm needs a pixel-mapped representation of a problem. For example, a map of cities in TSP is made to be discrete pixel pattern like Figure 7.

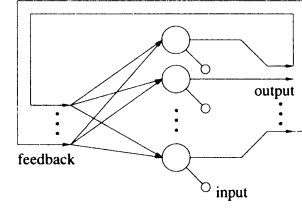


Figure 6: mutually connected neural network[8].

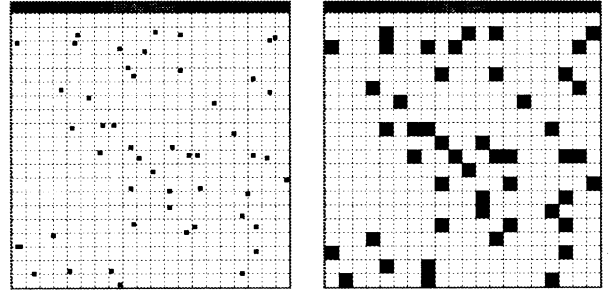


Figure 7: TSP map and its pixel representation.

3. Retrieve antibody using chaotic memory search using the similarity measurement:

$$C = \sum_{i=0}^{\frac{N}{P}-1} \mathbf{v}_{pi+q} \cdot \mathbf{v}'_{pi+q}$$

where  $p = \frac{\sqrt{P}}{2}$  and  $0 < q < p$ .

4. Generate initial population from the antibody.

Other steps of our algorithm are the same as original Immune Algorithm.

### 4 Experiments

Experiments are done for two problems each by two methods. The problems are 50 cities TSP and 50 items 0-1 Knapsack Problem. The methods are the original Immune Algorithm with randomly selected memory cells and the Immune Algorithm with neural adaptive memory. Table 1 shows conditions common with both methods. Subtour crossover method for TSP is used in Character-Preserving GA [9]. Table 2 shows conditions in adaptive memory search. In each experiment, test sets are given by slightly changing training set. Changing ratio 0% means that test set is equivalent to the training set.

Table 1: Conditions for each methods.

	TSP	Knapsack
number of antibodies	50	50
limit of generation	15000	15000
mutation rate	5%	5%
selection methods	elite strategy	elite strategy
crossover methods	subtour[9]	2-point

Table 2: Conditions for adaptive memory search.

number of pixels	20 × 20
population	10

The results of the experiments are shown in Figure 8. In both problem, IA with adaptive memory make a faster response than original IA with randomly selected initial population.

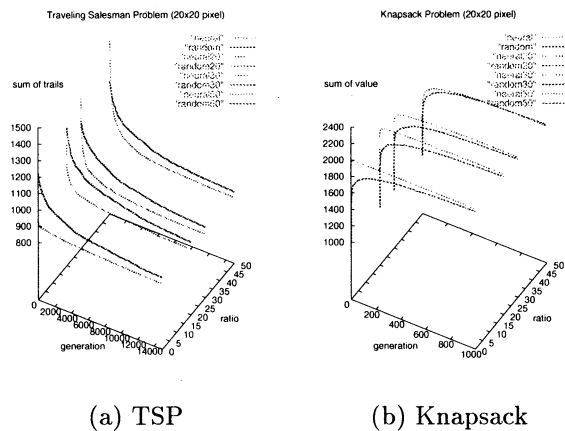


Figure 8: Results of experiments.

## 5 Conclusion

Immune Algorithm proposed by Mori et al. is so simple as to be applied to any problems which Genetic Algorithms is applicable to. However, this Immune Algorithm doesn't have enough ability to memorize antigen for future response.

Immune Algorithm with adaptive memory, proposed in this paper, has the same robustness as original IA, thus both have the same possible application. Furthermore, results of our simulations show that IA with adaptive memory are more efficient than original IA when solving similar problems repeatedly.

It seems that if new problem is entirely different from old problems, IA with adaptive memory might

make a slower response than original IA. It requires more researches to control this trade-off.

## Acknowledgments

This work was supported in part by Japan Society for the Promotion of Science grant (Research for the Future Program "Intelligence Information and Advanced Information Processing").

## References

- [1] H. Bersini, "Immune Network and Adaptive Control", *Proceedings of the First European Conference on Artificial Life*, ed. by F. Varela and P. Bourguin, MIT Press, pp. 217–226, 1991.
- [2] H. Bersini and F. Varela, "The Immune Learning Mechanism: Reinforcement, Recruitment and their Application", *Computing with Biological Metaphors*, ed. by R. Paton, Chapman and Hall, London, pp. 166–192, 1994.
- [3] E. Bonabeau, "Self-reorganization in a simple model of the immune system", *Physica A*, Vol. 208, pp. 336–350, 1994.
- [4] J. D. Farmer, N. H. Packard and A. S. Perelson, "The Immune System, Adaptation, and Machine Learning", *Physica D*, Vol. 22, pp. 187–204, 1986.
- [5] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.
- [6] K. Mori, M. Tsukiyama and T. Fukuda, "Immune algorithm with searching diversity and its application to resource allocation problem", *Transactions of the Institute of Electrical Engineers of Japan*, Vol. 113-C, pp. 872–878, 1993.
- [7] Y. Mori, P. Davis and S. Nara, "Pattern Retrieval in an Asymmetric Neural Network with Embedded Limit Cycles", *Physica A*, Vol. 22, pp. L525–L532, 1989.
- [8] S. Nara and P. Davis, "Chaotic Wandering and Search in a Cycle-Memory Neural Network", *Porg. Theor. Phys.*, Vol. 88, No. 5, pp. 845–865, 1992.
- [9] M. Yamamura, T. Ono and S. Kobayashi, "Character-Preserving Genetic Algorithms for Traveling Salesman Problem" (in Japanese), *Journal of Japanese Society for Artificial Intelligence*, Vol. 7, No. 6, 1992.

# Neurocontroller for load swing suppression of a jib crane on a floating bed

Fumihiro Tabuchi, Eiho Uezato, Hiroshi Kinjo and Tetsuhiko Yamamoto  
Department of Mechanical Engineering, Faculty of Engineering,  
University of the Ryukyus, Okinawa 903-0213, Japan

## Abstract

A jib crane on a floating bed is considered for the suppression of load swing. The jib crane operator controls the boom angle, boom length, rope length and swivel angle of the winch table. In this paper, we consider that the operator controls the boom angle or the boom length to suppress load swing, and discuss 2-dimensional motion of the crane system to simplify the problem. This paper proposes a new design method for automatic creation of a controller with the desired characteristics using a neural network evaluated by genetic algorithms (GAs). Because the proposed evaluations in GA operations are very simple and easy to understand, a machine designer who is not skillful in control theories is able to design the controller easily if he can program the simulator of the object system. The control value is the boom angle or the boom length. The control simulations show the validity of this method.

**Key words:** jib crane, neurocontrol, genetic algorithms, emergence, design method

## 1 Introduction

Many jib cranes are used at construction sites. Load swing caused by wind and/or boom actions disturb the effective working of the cranes. Several design methods<sup>1</sup> to control load swing have been proposed. Almost all of these methods require perfect comprehension of difficult control theories. In reality, most mechanical designers have not mas-

tered control theories. A new method which requires only basic knowledge of control theories will be a boon to busy designers.

A multi-input layered neural network acts as a multivariable nonlinear function, and can yield a nonlinear control gain in a crossed loop which includes dynamical systems. Here, we refer to a neural network which is considered as a controller, that is a "neurocontroller". Genetic algorithms evaluate the weights of a neural network to obtain the desired characteristics. The tasks remaining for the designer are decision of the evaluations for evolution of the neural networks and programming of the simulator of the object to be controlled. When the object plant is unstable, evaluations require dual steps(2) utilizing the responses of the object system. When the object plant is stable, the evaluation must be designed for rapid motion suppression.

## 2 Controlled System (Jib Crane)

The considered jib crane is illustrated in Fig. 1. The bed is supported by two spring-damper elements. The mass  $M$  and the inertial moment  $I$  of the bed are regarded as including the operator room and machine system (winch, engine, hydraulic system). The operator controls the boom angle  $\beta$  and/or the boom length  $r$  to suppress load swing. Let the operator be replaced by a neurocontroller. Figure 2 shows the free motion of the jib crane when  $\beta$  and  $r$  are fixed.



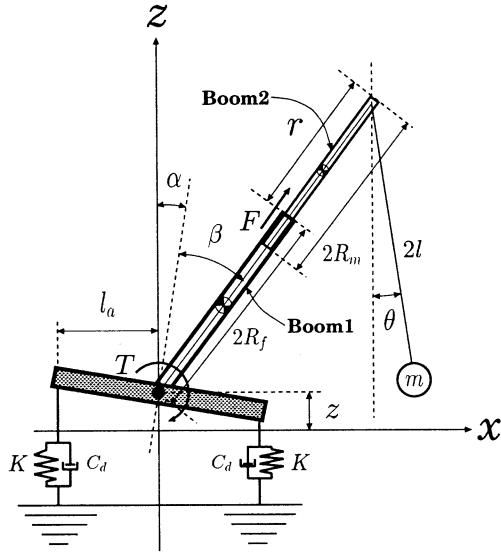


Fig. 1 Controlled element

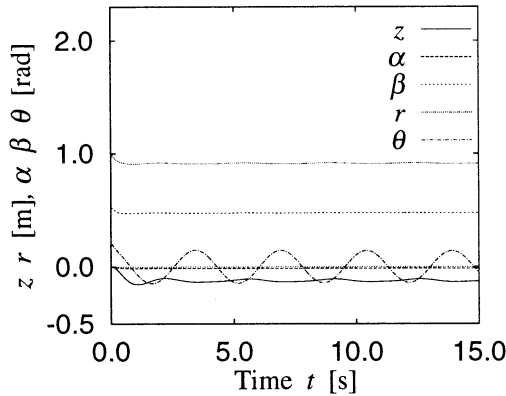


Fig. 2 Free motion of jib crane (noncontrolled)

### 3 Dynamics of Object System

- $z$  : displacement of bed
- $\alpha$  : inclination angle of bed
- $\beta$  : boom angle from the perpendicular line on the bed (controllable)
- $r$  : end jib length (controllable)
- $\theta$  : rope angle (pendulum swing angle)
- $l$  : rope length

### 4 Control System and Neurocontroller

Figure 3 shows the control system, where NN denotes the 10-input 15-element hidden layer 1-output neurocontroller, and GA denotes genetic algorithm procedures. The GA uses  $\mathbf{x}$  and  $d\mathbf{x}/dt$ , i.e., the responses of the object system, and calculates the weights of NN. NN refers to errors of the input  $\mathbf{x} (= [z, \alpha, \beta, r, \theta]^T)$  and the neurocontroller NN outputs the control value ( $= \dot{r}$  or  $= \dot{\beta}$ ). The system input (reference) of  $\theta$  equal to zero instructs the control system to suppress the load swing.

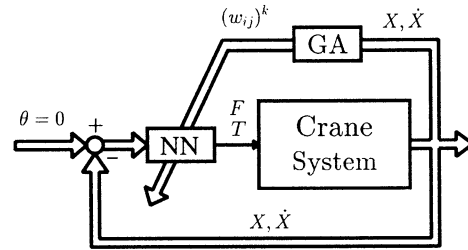


Fig. 3 Control system

### 5 Genetic Algorithm (GA) and Evaluations

GA is a powerful optimization technique, but the processing speed is dependent on the evaluation. A previous investigation<sup>2</sup> adopted the evaluation which utilized instability of the control object. In this paper we adopt the evaluation which utilizes stability of the control object. The number of neurocontrollers treated in the GA is 50. The ratios of crossover and mutation are 100 % and 2.0 %, respectively.

The procedure is shown in Fig. 4.

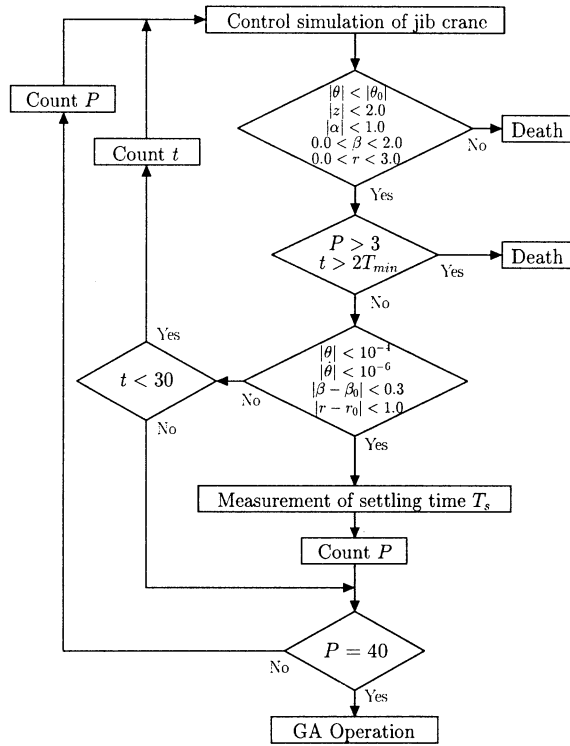


Fig. 4 Flow chart of evaluation

## 6 Evolution

The number of neurocontrollers which can safely suppress load swing increases with the generation process (see Fig. 5). The sudden decrease of the number of neurocontrollers is caused by the omission of the neurocontrollers which utilise long CPU times for settling to  $\theta = 0$ .

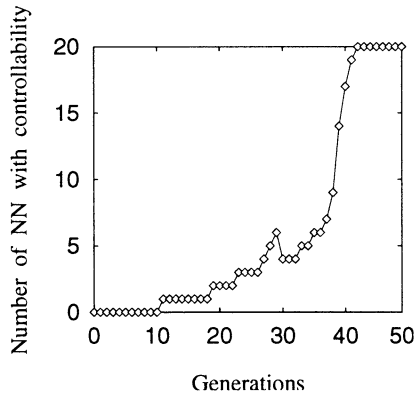


Fig. 5 Evaluation process

## 7 Control Simulation

The sufficiently developed neurocontroller simulates control of load swing suppression in a computer (see Fig. 6).

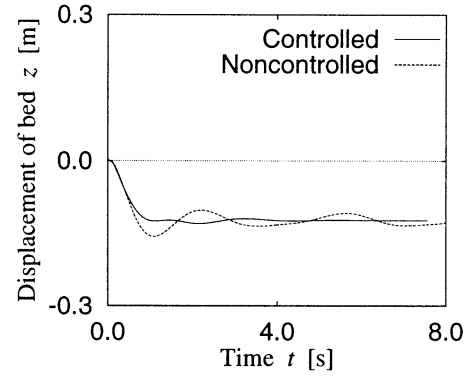


Fig. 6-1 Displacement of bed

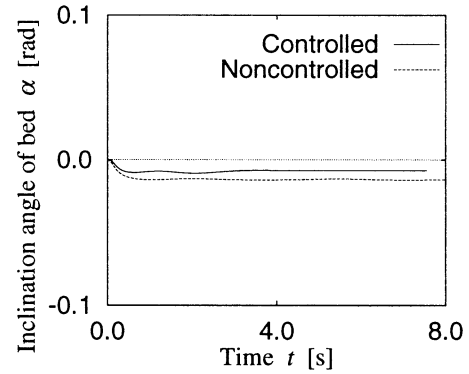


Fig. 6-2 Inclination angle of bed

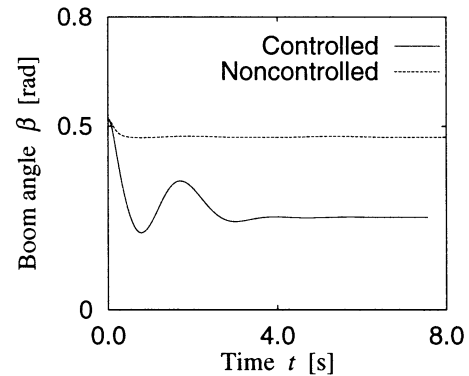


Fig. 6-3 Boom angle

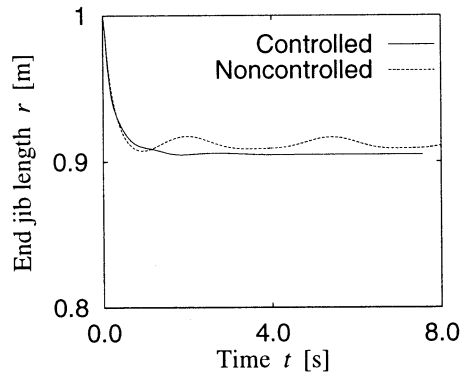


Fig. 6-4 End jib length

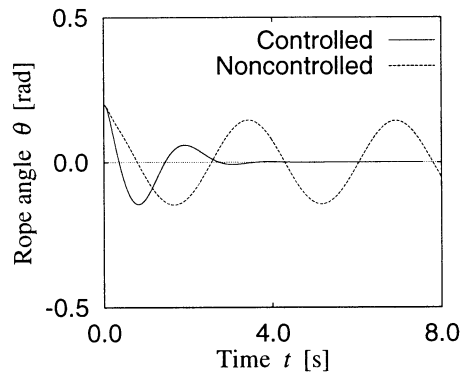


Fig. 6-5 Rope angle

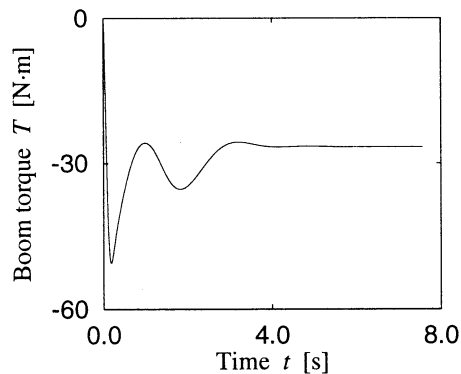


Fig. 6-6 Boom torque

presses the load swing by controlling the boom angle or the boom length. Neurocontrollers receive ten state variables and output a single force or torque value.

The evaluations in GAs are very simple because only the swing amplitudes and the settling time are measured and inspected. Measurements of the values mentioned above are faster than the calculation of the error integral.

## Acknowledgments

This research was partially supported by a Grant-In-Aid for Scientific Research (C) from the Ministry of Education, Science, Sports and Culture of Japan under Grant: Research No.09650295.

## References

1. For example, K.Hara, T.Yamamoto, A.Kobayashi and M.Okamoto; Jib crane control to suppress load swing, *Int.J.Sys.Sci.* Vol.20, No.5, pp.715-731, 1989
2. H.Kinjo, K.Nakazono, S.Tamaki and T.Yamamoto; Nonlinear Controller Using Neuralnet Tuned by GAs with Rough Evaluations, 2nd WCNA-96, July, 1996

## 8 Conclusion

Though a jib crane is a difficult object to control, the neurocontroller derived using GAs by means of the evaluation method explained in this paper sup-

# Self-organization of emotional neural network

Hiroshi KINJO, Hiroshi OCHI, Tetsuhiko YAMAMOTO  
Department of Mechanical Engineering, Faculty of Engineering,  
University of the Ryukyus, Okinawa 903-0213, Japan

## Abstract

In this paper an artificial emotional neural network is presented. For simplicity, two kinds of emotional neuron models are considered. One is a comfort neuron and the other is a discomfort neuron. The comfort neurons are activated by a comfort stimulus and the discomfort neurons are activated by a discomfort stimulus. The network is designed such that all emotional neurons strive to maintain the state of emotions in a moderate balance. If the emotional balance is upset for any reason, such as stress, then all of the emotional neurons start modifying network connections in order to recover the emotional balance. The modification method for the network connections is based on the Hebbian rule. The emotional neural network with a self-organized mechanism is demonstrated by numerical simulations.

**Key words:** Artificial neural network, Emotional neuron model, Self-organization, Autonomous adaptive systems

## 1 Introduction

Some actions of living organisms are related to emotions such as pleasure, anger, and fear. Emotions play one of the most important roles in the brain of living organisms. There are many papers related to the organization of neural networks and the effect of emotion on the brain<sup>1-8</sup>.

In this paper we present an artificial emotional neural network which consists of emotional neurons activated by an emotional stimulus from the envi-

ronment. We consider, for simplicity, two kinds of emotional neurons: one is a comfort neuron and the other is a discomfort neuron. The comfort neurons are activated by a comfort stimulus and the discomfort neurons are activated by a discomfort one. Furthermore, the comfort neurons are designed to be suppressed by the discomfort neurons and vice versa.

Balancing the neural activations between comfort and discomfort neurons is defined as the state of emotions in the network. The network is designed such that all of the emotional neurons strive to maintain the state of emotions in a moderate balance.

If the emotional balance is upset for any reason, such as stress, then all of the neurons start modifying the network connections in order to recover the emotional balance, by attempting to remove the stress signal. In this work, the modification method of the network connections is based on the Hebbian rule.

We show that an emotional neural network with a self-organization mechanism behaves as an autonomous adaptive system.

## 2 Emotional neural network

### 2.1 Comfort neurons and discomfort neurons

Figure 1 shows the emotional neuron model. The emotional neurons receive three kinds of input: from outside the network, from other emotional neurons, and from stress due to the environment.

The activation of the comfort neurons, written as

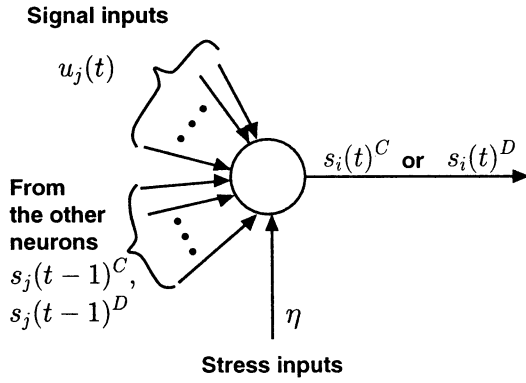


Fig. 1 Emotional neuron model.

$s_i(t)^C$ , is described by

$$s_i(t)^C = f(\text{net}_i(t)^C), \quad i = 1, 2, \dots, N_C \quad (1)$$

$$\text{net}_i(t)^C = \sum_{j=1}^{N_I} w_{ij} u_j(t) + \sum_{j=1}^{N_C} w_{ij} s_j(t-1)^C - \sum_{j=1}^{N_D} w_{ij} s_j(t-1)^D - \eta, \quad (2)$$

where  $f(\cdot)$  is the activation function of the emotional neurons.  $u_j(t)$  and  $w_{ij}$  are the input and connection weight, respectively.  $N_I$ ,  $N_C$  and  $N_D$  are the numbers of inputs, comfort neurons and discomfort neurons, respectively.  $s_j(t-1)^D$  denotes the activation of discomfort neurons.  $\eta$  is the stress signal described by

$$\eta = R(1 - \exp(-\alpha t)), \quad (3)$$

where  $R$  and  $\alpha$  are stress parameters. The activation of the comfort neurons is designed to be suppressed by the discomfort neurons and the stress signal.

In the case of discomfort neurons, the activations are described by

$$s_i(t)^D = f(\text{net}_i(t)^D), \quad i = 1, 2, \dots, N_D \quad (4)$$

$$\text{net}_i(t)^D = \sum_{j=1}^{N_I} w_{ij} u_j(t) - \sum_{j=1}^{N_C} w_{ij} s_j(t-1)^C + \sum_{j=1}^{N_D} w_{ij} s_j(t-1)^D + \eta. \quad (5)$$

A discomfort neuron is activated by other discomfort neurons and the stress signal.

## 2.2 Output neuron

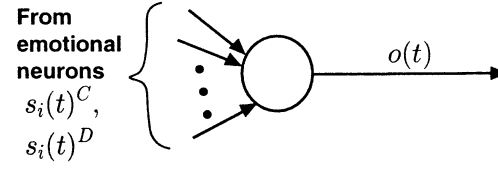


Fig. 2 Model of output neuron.

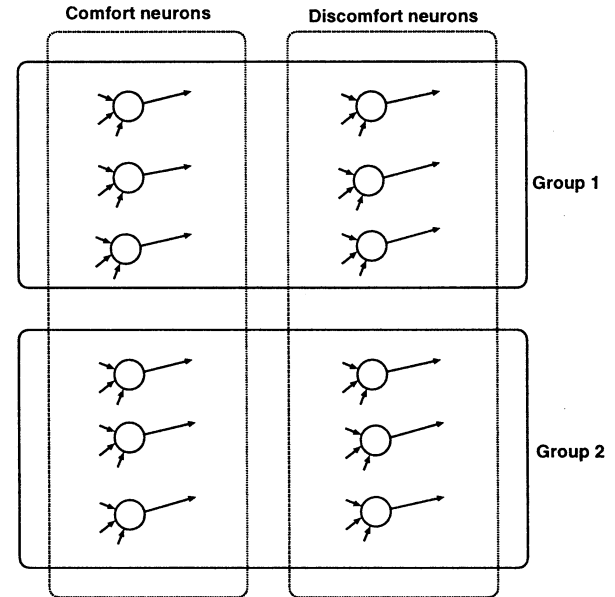


Fig. 3 Grouping of emotional neurons.

Figure 2 shows the model of output neuron. The output neuron receives all of the neural activations and selects on output of either zero or one. Figure 3 shows the grouping of the emotional neurons. Based on these groups, the output neuron selects the output value.

$$o(t) = \begin{cases} 0 & \text{for } a_1 \geq a_2 \\ 1 & \text{for } a_1 < a_2 \end{cases} \quad (6)$$

$$a_1 = \sum_{i \in \text{Group1}} s_i(t)^C + \sum_{i \in \text{Group1}} s_i(t)^D \quad (7)$$

$$a_2 = \sum_{i \in \text{Group2}} s_i(t)^C + \sum_{i \in \text{Group2}} s_i(t)^D \quad (8)$$

### 2.3 Emotional balance

We defined the state of emotions in the network by

$$P(t) = \sum_{i=1}^{N_C} s_i(t)^C - \sum_{i=1}^{N_D} s_i(t)^D. \quad (9)$$

The network is designed such that all of the neurons strive to maintain the emotional balance at a stable point.

## 3 Self-organization

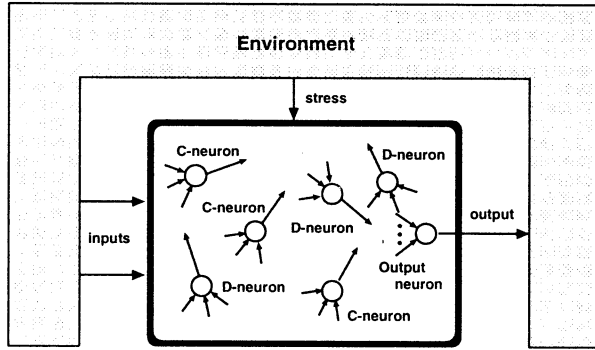


Fig. 4 Interaction between network and environment.

Figure 4 shows the interaction between the network and environment. The dynamics of the emotional neural network are as follows: when any signal is input to the network, the network outputs an arbitrary value to the environment. If, for example, the output is unsuitable for the environment, the network receives a stress signal such as a discomfort stimulus. Then the discomfort neurons are activated and repress the comfort neurons. As a result, the emotional balance is upset. At that time, all of the neurons start modifying the network connections in order to recover the emotional balance, by attempting to remove the stress signal. The modification method of the network connections is based on the Hebbian rule:

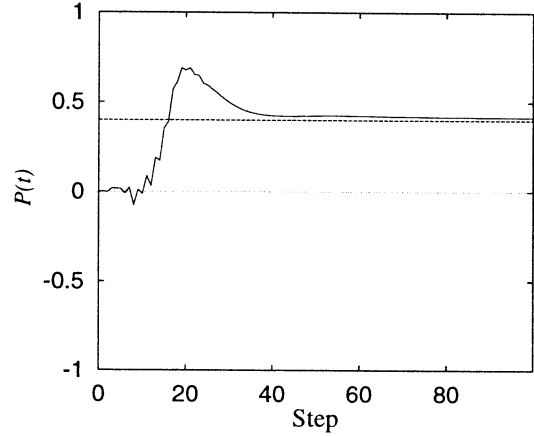
$$w_{ij}^{(\text{new})} = w_{ij}^{(\text{old})} + \varepsilon s_i(t) s_j(t) \quad (10)$$

$$\varepsilon = (P(t) - P_s)^2, \quad (11)$$

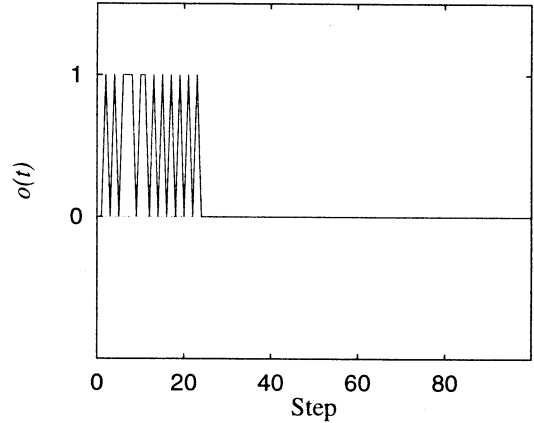
where  $P_s$  is a constant value corresponding to well-balanced emotions. Modification of the network

connections is continued until the emotional balance maintains the set condition and the stress signal is removed.

## 4 Numerical simulation



(a) Emotional balance



(b) Outputs

Fig. 5 Results of self-organization.  
(Environmental parameters were constant)

Table 1 Environmental parameters.

#	Input	Desired value
1	( 0 , 0 )	0
2	( 0 , 1 )	1
3	( 1 , 1 )	0
4	( 1 , 0 )	1

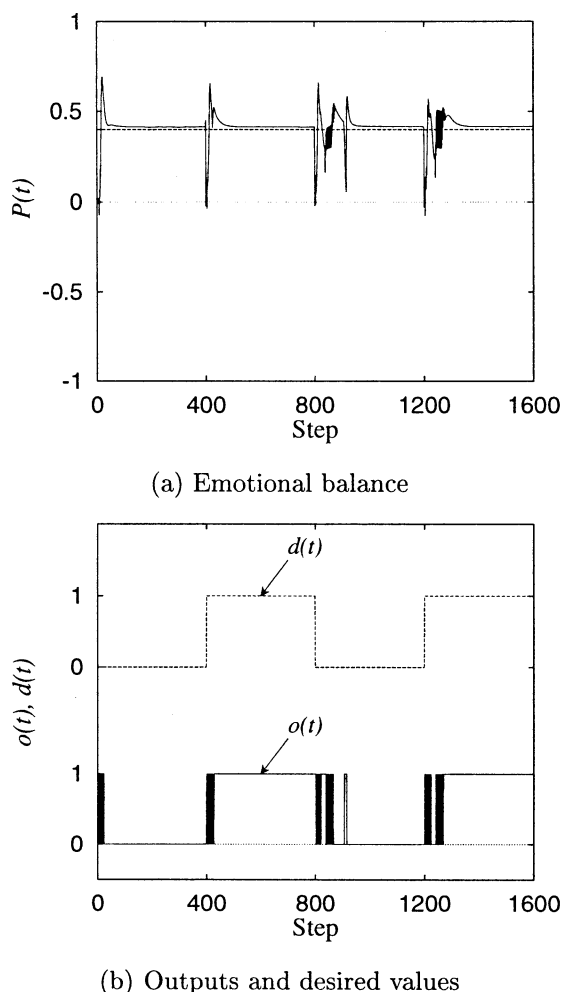


Fig. 6 Results of self-organization.  
(Environmental parameters were altered)

We consider an emotional neural network with two inputs and single output. The number of neurons is  $N_C = N_D = 6$ . The activation function of the neuron is a sigmoidal function. The stress parameters are  $R = 1.0$  and  $\alpha = 1.0$ .

In the simulation, signals (0, 0) were input to the network, at which time the desired value was set to zero. Figure 5 shows the results of the self-organization. In the figure, initially the network output changes significantly from zero to one and one to zero. Then the emotional balance rises slowly to the well-balanced point set to  $P_s = 0.4$ . After some steps, the emotional balance becomes the stable and the output of the network is zero, corresponding to the desired value.

Figure 6 shows the results of the simulation in which the environmental parameters altered from #1 to #4 in Table 1. It can be seen that in case of for various inputs and desired values, the emotional neural network attained the desired values.

## 5 Conclusion

In this paper we proposed an artificial neural network driven by an emotional stimulus from the environment. Simulations showed that the emotional neural network with a self-organization mechanism behaves as an autonomous adaptive system. The function of memory in the network and the stability of convergence are topics of future study.

## Acknowledgments

This research was partially supported by Grant-In-Aid No. 09650295 for Scientific Research (C) from the Ministry of Education, Science, Sports and Culture of Japan.

## References

1. D. O. Hebb (1949), The Organization of Behavior, New York, Wiley
2. J. A. Anderson, J. W. Silverstein, S. A. Ritz, and R.S. Jones (1977), Distinctive features, categorical perception, and probability learning: some applications of a neural model, Psychological Review, 84, pp. 413–451
3. A. Takeuchi and S. Amari (1979), Formation of topographic maps and columnar microstructures in nerve fields, Biological Cybernetics, 35, pp. 63–72
4. T. Kohonen (1984), Self-Organization and Associative Memory, Springer-Verlag
5. J. J. Hopfield and D. W. Tank (1985), "Neural" computation of decisions in optimization problems, Biological Cybernetics, 52, pp. 141–152
6. R. Linsker (1988), Self-organization in a perceptual network, Computer Magazine, 21, pp. 105–117
7. S. Tanaka (1990), Theory of self-organization of cortical maps: mathematical framework, Neural Networks, 3, pp. 625–640
8. H. Kitano (1995), Genetic algorithms II (in Japanese), Sangyo Tosho

# Intelligent Maneuvering of a Flight-Type Wall-Climbing Robot

Hiromori MIYAGI, Akira NISHI

Miyazaki University, Faculty of Engineering

Tel: 0985-58-2811 Fax: 0985-58-1647

Email: [miyagi@phys.miyazaki-u.ac.jp](mailto:miyagi@phys.miyazaki-u.ac.jp) [nishi@phys.miyazaki-u.ac.jp](mailto:nishi@phys.miyazaki-u.ac.jp)

The manuscript of this talk will be distributed at the symposium.



# Block Fuzzy Neural Networks for controlling an Industrial Manipulator

Jun Tang and Katsutosi Kuribayashi

Department of Mechanical Engineering, Yamaguchi University,  
Tokiwadai-2557, Ube 755-8611, Japan

**Keywords:** Robot Manipulators, Control Method, Fuzzy Neural Networks, Tracking Control

## Abstract

This paper discusses trajectory tracking control of an industrial manipulator. A new control algorithm is proposed based on the block fuzzy neural network method. The block controller is realized by using several double-input and single output or single-input and single-output fuzzy gaussian neural network. This paper describes that the traditional control system is resolved into several basic control blocks according to its control aim. Each control block is constructed by a rigorous fuzzy reasoning, which is assumed to be a simplified reasoning. Some experimental test results are also included to demonstrate the improvement in the tracking performance when the proposed method is used.

## 1. Introduction

It is well known that the industrial manipulators are uncertain nonlinear dynamic systems which suffer from structures and uncertainties on the inertia parameters[1]. Most current industrial manipulators are equipped with these simple controllers such as PD or PID, which have been proved to be effective position or tracking controllers[2][3]. Despite the extensive spread of the simple control in robotic manipulators, it is relatively difficult to tune the control gains with respect to varying desired configurations when the controller is designed, and it is also unknown for the tuning aim of the control gains, because the trajectory tracking control for the industrial manipulator is the complex ill-defined processes. For these reasons, we need to consider a simple and effective control method to solve the known problem of the tuning gain.

In the past several years, fuzzy control systems have the advantages of being robust to nonlinear system

and the variations of system dynamics. They have been successfully utilized in the complex ill-defined processes with better performance than that of PID controllers. However, there are still difficulties in the design of fuzzy controllers. In particular, the number of fuzzy rules increases exponentially as the number of input variables of fuzzy controllers increases for multi-input/multi-output. The goal of this paper is to present a block fuzzy neural network controller, which can solve the problems above. It is also applied to control an industrial manipulator. Because the controller can be tuned individually according to the control purpose of each block, the tuning of the system is changed into a known process, compared to the traditional control system.

The paper is organized as follows. Section 2 describes the dynamics of the industrial robotic manipulator, including its kinematics and Jacobian matrix. In section 3, we review the control law of the PID controller with gravity compensation, and discuss the features of the controller in the real-time control of the manipulator. In section 4.1, we describe the fuzzy neural network that uses a single-input and single-output neural network. Then we present the block fuzzy neural networks controller, and apply it to control an industrial manipulator. A Jacobian neural network is also discussed for the trajectory tracking control. Experimental results on tracking control of the manipulator are presented in section 5. Section 6 draws some conclusions from the work.

## 2. Robot Dynamics

Consider the vector representation of an  $n$ -link rigid manipulator dynamics given by [4]

$$\tau = M(\theta)\ddot{\theta} + h(\theta, \dot{\theta})\dot{\theta} + g(\theta) \quad (1)$$

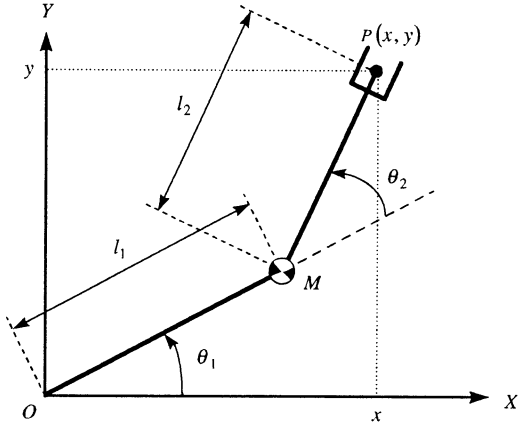


Figure 1: 2-link manipulator model

where  $\theta$  is the  $n \times 1$  vector of joint displacements,  $\dot{\theta}$  is the  $n \times 1$  vector of joint velocities,  $\tau$  is the  $n \times 1$  vector of applied torques,  $M(\theta)$  is the  $n \times n$  symmetric positive-definite manipulator inertia matrix,  $h(\theta, \dot{\theta})$  is the  $n \times n$  matrix of centrifugal and Coriolis forces, and  $g(\theta)$  is the  $n \times 1$  vector of gravitational torques obtained as the gradient of the robot potential energy  $\gamma(\theta)$ , i.e.,

$$g(\theta) = \frac{\partial \gamma(\theta)}{\partial \theta} \quad (2)$$

Here, the model of a two-link robotic manipulator is shown in Fig. 1 and its dynamic equations in the absence of loads and friction can be written as:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \ddot{\theta} + \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} \dot{\theta} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad (3)$$

where  $\theta \triangleq [\theta_1 \ \theta_2]^T$ . The kinematics equation is also given by

$$x = l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) \quad (4)$$

$$y = l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) \quad (5)$$

where the joint angles are defined by  $q = (\theta_1, \theta_2)$ ,  $l_1$  and  $l_2$  are the length of the link 1 and link 2, respectively.

Letting the end-point position of robot be defined by  $p = (x, y)$ , it is well-known that the joint and end-point velocity vectors satisfy

$$\dot{p} = J\dot{q} \quad (6)$$

where  $J$  is the Jacobian matrix expressed by

$$J = \begin{bmatrix} -l_1 S_1 + l_2 S_{12} & -l_2 S_{12} \\ l_1 C_1 + l_2 C_{12} & l_2 C_{12} \end{bmatrix} \quad (7)$$

Here,  $S_1 = \sin \theta_1$ ,  $S_{12} = \sin(\theta_1 + \theta_2)$ ,  $C_1 = \cos \theta_1$ , and  $C_{12} = \cos(\theta_1 + \theta_2)$ .

### 3. PID Controller with Gravity Compensation

Consider the relation of the joint torque  $\tau$  and the force applied to the end-point position  $f$  [4], which is given by

$$\tau = J^T(\theta) f \quad (8)$$

Then the following PID controller with gravity compensation is obtained from (8):

$$\tau = J^T(\theta)(K_P e - K_D \dot{e} + K_I \int e dt) + g(\theta) \quad (9)$$

where  $K_P$ ,  $K_I$  and  $K_D$  are the controller parameters, respectively, the proportional, integral and derivative gain matrices. The tracking error of the end-point position is defined by

$$e \triangleq p_d - p \quad (10)$$

with  $p_d = [x_d(t) \ y_d(t)]^T$  being the vector of a desired position. The PID control law with gravity compensation uses the gravitational torque vector, it solves the problem of varying gravity of the industrial robot. For the tracking problem of a desired time-varying trajectory, the design of the control system is considered to deal with the varying velocities of the end-point position, because for such a case the joint velocities are not directly needed to be regulated by using a D-control with the joint velocity error.

## 4. Block Fuzzy Neural Network System

### 4.1 Fuzzy Neural Network (FNN)

The fuzzy reasoning adopted here is a simplified reasoning method, which can be interpreted as a special case of the Sugeno's fuzzy reasoning. This method coincides with the case when the function in the conclusion of the Sugeno's fuzzy reasoning becomes a constant, or when the width of the fuzzy set in the conclusion of the min-max-centroidal method becomes an infinitesimal value, that is, a singleton. Therefore, any  $i$ -th control rule can be written by

$$\begin{aligned} R_i : & \text{ If } x_1 = A_{i1} \text{ and } \dots \text{ and } x_n = A_{in} \\ & \text{ then } u_1 = B_{i1} \text{ and } \dots \text{ and } u_p = B_{ip} \end{aligned} \quad (10)$$

where  $R_i$  denotes the  $i$ -th control rule,  $A_{ij}$  the fuzzy set in the antecedent associated with the  $j$ -th input variable at the  $i$ -th control rule,  $B_{ij}$  denotes a constant associated with the  $j$ -th variable in the conclusion at the  $i$ -th control rule. Applying  $n$  confidences  $\mu_{A_{i1}}(x_1), \dots, \mu_{A_{in}}(x_n)$ , the confidence in the antecedent  $h_i$  is defined by

$$h_i = \mu_{A_{i1}}(x_1) \cdot \mu_{A_{i2}}(x_2) \dots \mu_{A_{in}}(x_n) \quad (12)$$

where “ $\cdot$ ” is the algebraic product. Then, the  $j$ -th output consequent can be calculated as the following weighted mean of  $B_{ij}$  with respect to the weight  $h_i$ :

$$u_j^* = \frac{\sum_{i=1}^r h_i B_{ij}}{\sum_{i=1}^r h_i}, \quad j = 1, \dots, p \quad (13)$$

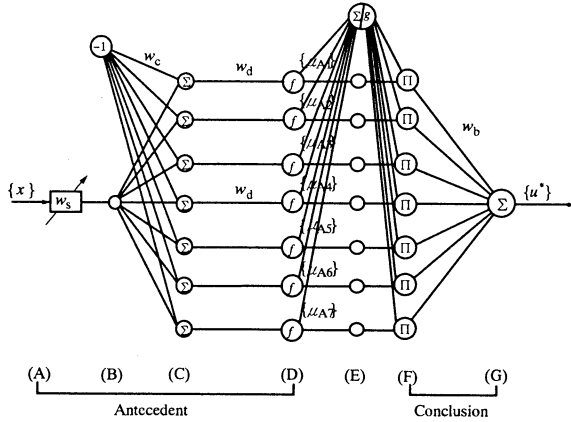
The FNN method uses the following Gaussian function as a unit function [5]:

$$f(x) = e^{\ln(0.5) \cdot x^2} \quad (14)$$

For this case, the construction of the membership function using a neural network is depicted. Figure 2 illustrates a construction example of the FGNN for a case where there are two inputs  $(x_1, x_2)$ , single output  $(u_1^*)$  and three labels in the antecedent part. Then, the number of identifiable control rules is  $r = 3^2$ . For the F layer in the figure, the unit that has symbols  $\Sigma$  and  $g$  generates the output through the following function:

$$g(x) = \frac{1}{x} \quad (15)$$

with a linear summed input. The layers A~E in Fig. 2 correspond to the antecedent part of the fuzzy control rule, and the layers G and H correspond to the conclusion part. In addition,  $w_s$  denotes the input scaler and the connection weight  $w_b$  is the constant values in the conclusion part.



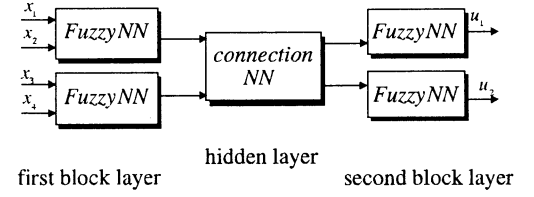
**Figure 2:** Fuzzy neural network with two-input and single-output

#### 4.2 Block FNN Control System

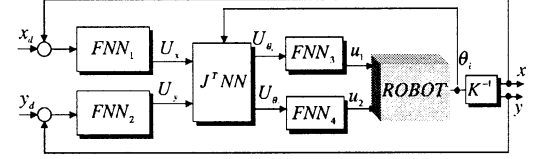
The block FNN shown in Fig. 3, it is composed the first block layer and the second block layer. Here, a FNN is regarded as a block unit. Note that we also set a hidden layer which connects the first and second layers. The inputs and outputs of the block fuzzy controller are assumed to be  $\{x_1, x_2, x_3, x_4\}$  and  $\{u_1, u_2\}$ , respectively. Its first block layer is comprised of two FNN with two-inputs and single-output, and the second one is two FNN with the single-input and single-output.

If the error of the control system is defined as  $e = [e_x(t) \ e_y(t)]^T$ , we can propose the following assumptions now;

- A1. In the first layer block as a input layer, the reasoning torques  $\{U_x, U_y\}$  for controlling the end-point positions of the robot are generated from the X-position errors  $e_x$  and



**Figure 3:** The block fuzzy neural networks



**Figure 4:**Block diagram of the block fuzzy controller

its rate  $\dot{e}_x$ , and the Y-position errors  $e_y$  and its rate  $\dot{e}_y$ .

- A2. In the second layer block as a output layer with the reasoning inputs  $\{U_{\theta_1}, U_{\theta_2}\}$ , the actual driving joint torques  $\{u_{\theta_1}, u_{\theta_2}\}$  for the two links are generated.

Note that the relationship between the torques of the end-point positions and the driving joint torques were given in (8). As shown in Fig. 4, the block diagram of block FNN controller is depicted.

#### 5. Examples of Simulation

In this section, some computer simulations are conducted to demonstrate the performance of above method. The manipulator was modeled as a nonlinear industrial robotic manipulator, whose parameters are as follows:

$$l_1 = 270 \text{ [mm]}, \quad l_2 = 450 \text{ [mm]}, \quad l_{g1} = 185 \text{ [mm]}$$

$$m_1 = 5.68 \text{ [kg]}, \quad m_2 = 4.17 \text{ [kg]}, \quad l_{g2} = 147 \text{ [mm]}$$

$$I_{i1} = 10.1 \text{ [kgm}^2\text{]}, \quad I_{i2} = 2.756 \text{ [kgm}^2\text{]}$$

Let us now consider a tracking control problem of a circular trajectory with 0.2 [m] diameter. The following desired trajectories in X- and Y-axes of the circle was considered:

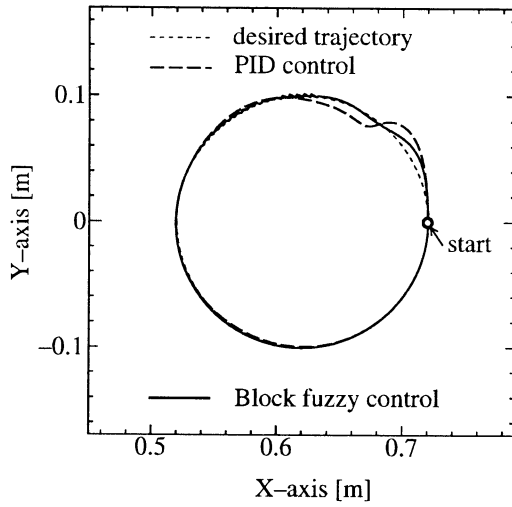
$$x_d(t_i) = 0.62 + 0.1 \cos(2\pi\lambda) \quad (17)$$

$$y_d(t_i) = 0.1 \sin(2\pi\lambda) \quad (18)$$

where the experimental time was set to be  $t_N = 20$  [s] and the control sampling period was  $T = 25$  [ms],  $\lambda = -2(i/T)^3 + 3(i/T)^2 \quad i = 1, \dots, N$ .

At first, we use an original method (the fixed PID control) to control the robot. The fixed PID control gains were chosen by trial and error as follows:

$$K_{Px} = 210.0, \quad K_{Ix} = 19.0, \quad K_{Dx} = 19.0$$

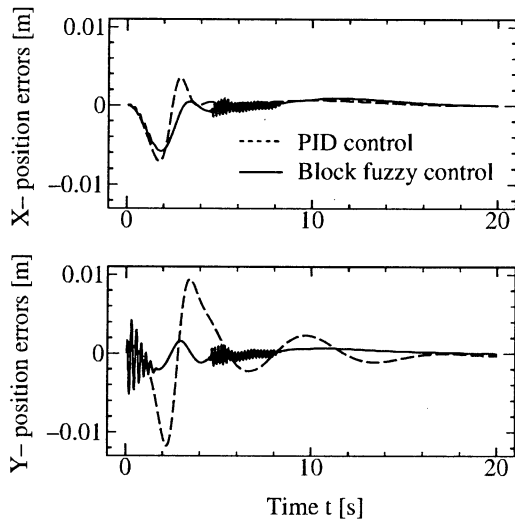


**Figure 5:** Trajectory control results using the PID and block fuzzy controller

$$K_{Py} = 30.0, \quad K_{Iy} = 15.0, \quad K_{Dy} = 16.0$$

Note here that this gains were obtained only using the experience or trial and error. It is not easy for us to find out the tuning aim in the experiment. Figure 5 shows the result of the tracking trajectory. As can be seen from this figure, the desired trajectory for the X- and Y-motion axes is achieved, though there is a big trajectory error at the transient response due to the robotic inertia. However, the tracking errors are given by  $|e_x| \simeq 0.05$  [m] and  $|e_y| \simeq 0.01$  [m] as seen from Fig. 6.

To illustrate the above-proposed method, we compare the fixed PID controller with the proposed block fuzzy control law. In the case of the block fuzzy control, in order to automate the parameter tuning of fuzzy controller, an adaptive input scaling technique



**Figure 6:** Tracking errors for X- and Y-axes

is applied to determine the input scalars [5], and the conclusion constants  $B_{ij}$  is designed as a switching line of the VSS controller [6].

For two-inputs and single-output first block FNN, the input scalars were selected as 20.0, 80.0, 2.5, 1.4. The constant values in the conclusion were calculated using  $\alpha = 1.0$  and  $\beta = 2.0$  as shown the reference [5], where  $|w_{bi}| \leq L (= 6)$ . Here, the center values of 5 labels,  $w_c$ , were  $-6, -3, 0, 3, 6$ , and the reciprocal values of deviation  $w_d$  were all unity in order to allocate all labels with a equal space on the support set  $[-6, 6]$ . For the single input and single output second block FNN, the center values of 7 labels,  $w_c$ , were set as  $-6, -4, -2, 0, 2, 4, 6$ , and the control rules,  $w_b$ , were  $-0.6, -0.4, -0.2, 0, 0.2, 0.4, 0.6$ . Note that the tuning aim is assumed to be the actual driving joint torques. From Fig. 5, by using block fuzzy control, the control performance is improved than that of the fixed PID control. It is seen that the proposed method is faster and smaller overshoots in comparison to that of PID (see Fig. 6).

## 6. Conclusions

A block fuzzy controller is proposed in this paper. We adopt four independent fuzzy controllers consisted of a three layers block hierarchical controller and apply it to controlling an industrial manipulator. The proposed method is shown to work well in the control of the robot. Some actual experiments are being done by using the proposed method. We will show the results in this conference.

## References

- [1] S. Nicosia and P. Tomei, "Robot Control by Using only Joint Position Measurements," *IEEE Trans. on Automatic Control*, Vol. **AC-35**, pp. 1058-1061, 1990.
- [2] P. Tomei, "Adaptive PD Controller for Robot Manipulators," *IEEE Tran. on Robotics and Automation*, Vol. **7**, No. 4, pp. 565-570, 1991.
- [3] Canudas de Wit, C., Fixot, N and Astrom, K., "Trajectory Tracking in Robot Manipulators via Nonlinear Estimated State Feedback," *IEEE Trans. on Robot and Automation*, Vol. **8**, No. 1, pp. 138-144, 1992.
- [4] C. Canudas de Wit, B. Siciliano, and G. Bastin, *Theory of Robot Control*, Springer Press, 1996.
- [5] J. Tang, K. Watanabe, and M. Nakamura, "Learning Control of a Mobile Robot Using Block Hierarchical Fuzzy Gaussian Neural Networks," *Japanese Journal of Fuzzy Theory and Systems*, vol. **8**, no. 1, pp. 85-98, 1996.
- [6] K. Watanabe, K. Hara, and S.G. Tzafestas, "Fuzzy Controller Design Using the Mean-Value-Based Functional Reasoning," *Procs. of IJCNN'93*, no. 3, pp. 2983-2986, 1993.

## A Fuzzy Model to Control the Temperature in Cooling of Metal Molds with Spray Robot

*Tatsuo Sakamoto, Katsutoshi Kuribayashi\*, Kousei Murakami*

Ube Industries Co., Ltd. Machinery Div.  
1981 Okinoyama, Ube, 755 Japan

\*Yamaguchi University, Faculty of Engineering  
2557 Tokiwadai, Ube, 755 Japan

### Abstract

In die casting operation, die spray is the key for the quality and for the productivity. But die spray operation depends its parameter setting such as spray volume and path on operator's experiences even the robot sprayer is adopted.

The purpose of this study is to establish an intelligent system to control surface temperature of die with water base mist spray.

A fuzzy model was applied to determine optimum mist spray volume. The generated results show good matching with actual measurement regarding the temperature drop rate versus spray mist volume.

After the fuzzy analysis has been done, by using the calculated spray parameters, experiments to control actual die temperature conducted on the heated plate model.

The fuzzy model constructed by using fundamental experiments allows die spray operation with predetermined spray volume.

**Key words:** Fuzzy Model, Die Cast, Spray Robot  
Die Temperature, Spray Flow Rate

### 1. Introduction

Die Casting is the most effective production process to make net shape parts for use in automobiles, appliances, etc. This process is the high pressure casting which injects the molten metal into the die cavity and quickly solidify. Die spray is the key for this process. The spray provides a cooling effect, thermal insulation, and aids the release of the part from the die. The difficulty of spraying operation is to maintain or to adjust die surface temperature in desired level by setting the mist volume suitable on the various cavity geometry<sup>(1)(2)</sup>. Therefore spray parameters settings such as spray mist volume and path are deeply depend on operators experiences.

The purpose of this study is to obtain water mist volume to adjust the die surface temperature to the desired level, even the die surface temperature and spray nozzle movement are varies according to die cavity configurations.

Assuming the cooling effect is obtained by evaporation of water base die lubricant mist spray, the cooling rate will be described as a function of initial die surface temperature and spray mist volume.

And the unit volume of spray mist varies not only by volume control but with spray nozzle moving speed and distance between die surface also.

Therefore, a fuzzy model was set for 4 parameters:

- (1) Initial die surface temperature
- (2) Spray mist volume
- (3) Distance between nozzle and die surface
- (4) Nozzle moving speed.

The calculated spray volume with fuzzy model was conveyed to the spray robot to demonstrate spray on hot die with temperature distribution.

### 2. Preparation of the Experiments

The basic cooling rate measurement and fuzzy analysis have been done on the flat steel die, sized 370mm(H) x 350mm(W).

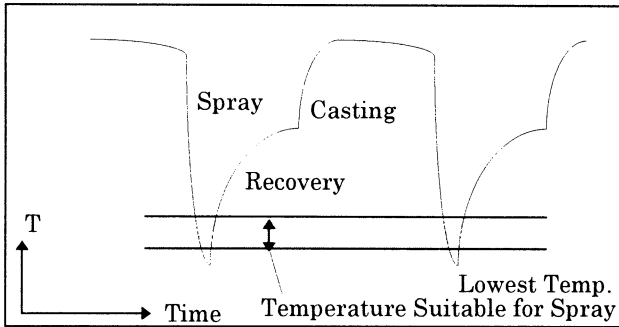
To experiment spray operation with three parameters (2),(3),(4) as described in previous section, a 5-axis robot and 1-axis servo control water discharge pump were prepared.

At the experiments, die surface temperature was measured with infrared thermograph and the temperature distribution images are stored in the personal computer in every 30 mili-second.

As shown in [Fig.1-1], then the temperature drop was measured by the lowest temperature reading which die surface reaches with spray, before it is recovered with heat mass stored in the die.

This measurement criteria was decided because the die temperature control with mist spray is for the suitable temperature range to the spray

agent sticks on the die surface. Actually after few seconds die surface temperature will be recovered with heat from inside of the die.



[Fig. 1-1] Temperature Profile on Die Surface

### 3. Fuzzy model of this study<sup>(3)</sup>

The model proposed in this study is 4-input single output model. The procedure followed to construct the model consists of three steps:

- 1) Fuzzification of inputs
- 2) Construction of the influenced engine
- 3) Defuzzification of the outputs.

#### 3.1 Fuzzification of the input

Fuzzification is the process of generating membership values for a fuzzy variable through membership functions. In this step the fuzzification of the inputs was carried out based on the spray conditions. This data was obtained by experiments carried out during operations of the spray robot, as described in section 2.

Based on this data the membership functions of the model were made as [Fig 3.1].

The fuzzified input parameters are:

- Temperature -Die surface temperature in  $^{\circ}\text{C}$
- Spray distance -Distance between die surface and spray nozzle in mm.
- Spray flow rate -Spray volume in cc/s.
- Velocity -Moving velocity of the robot in mm/s.

The tables ( 3.1 to 3.4 ) show the data obtained by the experiments. Each table shows the influence of each variable to the temperature reduction (output) keeping the other variables as constant parameters. They show also the linguistic variables chosen as membership functions for the model.

Name of the membership		high	average	low
Temperature	$^{\circ}\text{C}$	400	300	200
Spray distance	mm	200	←	←
Spray flow rate	cc/s	18	←	←
Velocity	mm/s	300	←	←
Temperature reduction	$^{\circ}\text{C}$	200	140	120

[Table-3.1] Influence of the initial temperature

Name of the membership		high	average	low
Spray distance	mm	100	200	300
Temperature	$^{\circ}\text{C}$	300	←	←
Spray flow rate	cc/s	18	←	←
Velocity	mm/s	300	←	←
Temperature reduction	$^{\circ}\text{C}$	170	140	110

[Table-3.2] Influence of the spray distance

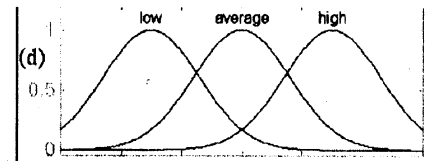
Name of the membership		high	average	low
Spray flow rate	$^{\circ}\text{C}$	36	18	6
Spray distance	mm	200	←	←
Temperature	cc/s	300	←	←
Velocity	mm/s	300	←	←
Temperature reduction	$^{\circ}\text{C}$	190	140	100

[Table-3.3] Influence of the spray flow rate

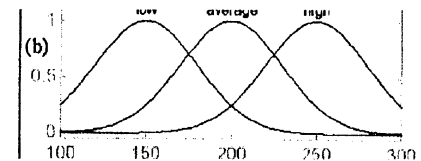
Name of the membership		high	average	low
Velocity	$^{\circ}\text{C}$	300	600	900
Spray distance	mm	200	←	←
Spray flow rate	cc/s	18	←	←
Temperature	mm/s	300	←	←
Temperature reduction	$^{\circ}\text{C}$	140	130	120

[Table-3.4] Influence of the Nozzle Velocity

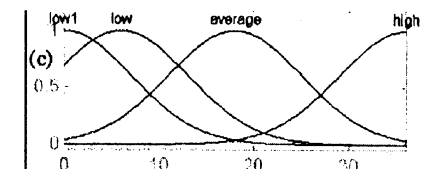
Membership Functions for the Initial Temperature



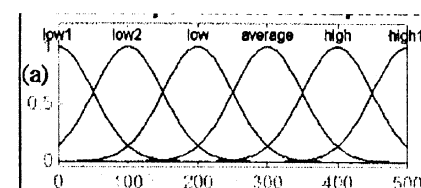
Membership Functions for the Distance



Membership Functions for the Flow Rate



Membership Functions for the Temperature



[Fig.3-1] Membership Functions

#### 3.2 The Inference Engine

The inference engine is the fuzzy reasoning (rule-based reasoning) used to map an input space to an output space, that is, the part of the model where the rules to control the system are placed. These rules were elaborated according to

the data of the spray conditions.  
[Fig. 3-2] is the rule of the model.

1. if ( $T$ is $L$ ) & ( $D$ is $M$ ) & ( $F$ is $M$ ) & ( $V$ is $L$ ) then ( $\Delta T$ is $mf1$ )
⋮
12. if ( $T$ is $M$ ) & ( $D$ is $M$ ) & ( $F$ is $M$ ) & ( $V$ is $L$ ) then ( $\Delta T$ is $mf12$ )
$T$ = Die surface initial temperature
$D$ = Distance between die and spray nozzle
$F$ = Flow rate of the spray liquid
$V$ = Velocity of the nozzle movement
$\Delta T$ = Temperature drop
H, M, L = High, Medium, Low, respectively

[Fig. 3.2] Fuzzy rule of the model

### 3.3 Defuzzification

Defuzzification is the process designed to produce a non-fuzzy output or control action that optimally represents the possibility distribution of an inferred action. In this research the output to be controlled is ( $\Delta T$ ), the temperature reduction. For a determined spray condition the model calculates the temperature reduction and it also calculates the spray flow rate for a set up temperature reduction. The functions mf1 to mf12 shown in [Fig.3.2] represent these singleton functions.

The values estimated for the mfs are, for example:

mf1 =110.3 in measurement 120  
mf2 =140.0 in measurement 140  
mf6 =172.2 in measurement 170  
mf12=122.5 in measurement 130.

## 4. Result of Simulations Using the Model

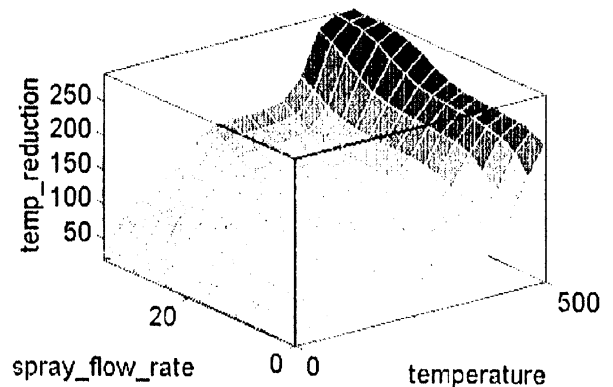
### 4.1 Simulation

The simulations of the model were performed using high-performance simulation software for numeric computation and visualization MATLAB. Using the Fuzzy Logic Toolbox of this software, the fuzzy inference system for this study was built. Firstly, the membership functions and the rule-based controller were built and then the methods of fuzzy inference functions were adjusted. Secondly, the fuzzy model was integrated into a block diagram using SIMULINK, that is a tool to build control models using block diagrams for simulations, to evaluate the behavior of the parameters of the model in the simulations. Finally, numeric data was obtained using the model. For given input parameter values (temperature, spray distance, spray flow rate and moving velocity of the robot)

the output parameter values (temperature reduction) were calculated using command line functions.

### 4.2 Results of the Simulations

The results of the simulations were given in 3 different ways, in 3D graphics, in 2D graphics and numeric result. [Fig.4-1] shows one of 3D outputs.



[Fig.4-1] 3-D Output

Numeric results were calculated based on the parameters for which the practical use of experiments. The temperature reduction  $\Delta T$  was calculated for given input value of the initial die temperature for spray flow rate from 0 to 36 cc/s with the parameters spray distance and velocity as constants in the values 200mm and 300mm/s respectively.

## 5. Experiments and Result

### 5.1 Experimental Procedure

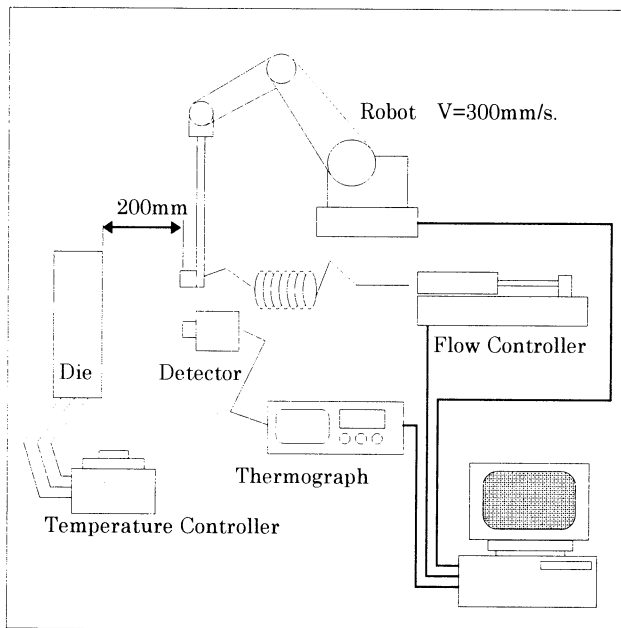
In order to evaluate the accuracy of the data calculated through the fuzzy model, experiments were conducted in which it compared the temperature distributions before and after spray on a die surface. The spray mist is applied on the different mesh surface of the die at different flow rates. The optimum flow rate for a desired temperature reduction at each mesh was calculated through the model according to its respective temperature averages. A infrared thermograph was employed to obtain the temperature data. This experimental procedure was as follows:

- A pre-experiment was carried out to obtain the data of the temperature distribution on the entire surface of the die (before spray). This data is the image of temperature distribution of the hot surface of the die and can be saved as digital data. Then, based on this image data, the surface of the die was divided

into 16 different meshes of the same size area and the respective temperature average for each mesh was calculated.

- (b) Based on the temperature average of each mesh, the optimum flow rate for a desired temperature reduction on each mesh was calculated by the model. A 5-axis spray robot with one axis flow control pump were used. Spray mist was applied at the 16 different meshes at different flow rates, according to the respective temperature average. Then, the image data of the temperature distribution was obtained.

Fig.5-1 is the arrangement for the experiments.



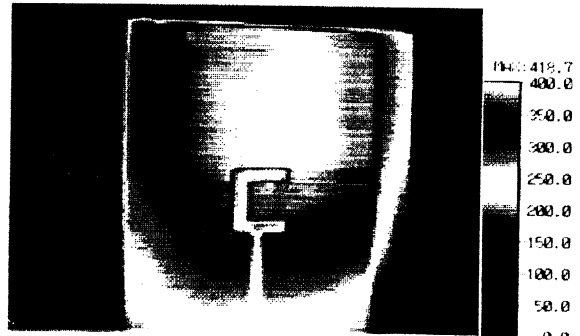
[Fig. 5-1] Experimental Arrangement

## 5.2 Experimental result

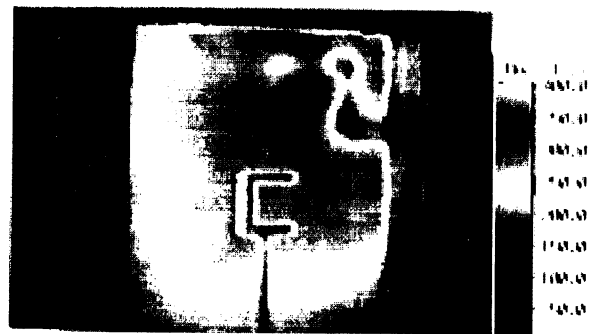
Based on the temperature average of each mesh, spray flow rates for uniform temperature distributions on the surface of the mold at 210° C were calculated by the fuzzy model. The spray robot performed a path delivering different flow rates at each mesh in 5 seconds. The image data of the temperature distribution was taken by the thermograph. It saved image frame in every 30 milli-second during the performance of each spray operation.

In the experiments, the steam was generated during the spray operation it masked the real image of the temperature distribution which was taken by the thermograph, Figs.[5.2] and [5.3] show the operation performed by the spray robot. Analyzing samples of meshes through the image frames obtained by the thermograph, in some samples of meshes which does not covered with

steam, it was observed a considerable temperature reduction seems similar to the predetermine values. But for the true evaluation, the method to eliminate steam influences is required.



[Fig. 5-2] Temperature Distribution before the Spray Operation



[Fig. 5-3] Temperature Distribution during the Spray Operation

## 6. Conclusion

As a result of this study the following was accomplished.

- (1) The 4-single input fuzzy model was designed and usefully used to estimate temperature drop of the die.
- (2) The fuzzy model were constructed by using fundamental experiments.
- (3) The optimum spray volume to control the die surface temperature was calculated by the fuzzy model.

## Reference

- (1) A. Barbed de Magalhaes, Ricardo Paiva  
Intelligent System to Control the Temperature of Permanent Molds and Dies  
62<sup>nd</sup> World Foundry Congress- Philadelphia
- (2) Society of Manufacturing Engineers:  
Tool and Manufacturing Engineers Handbook  
NacGraw-hill, 1976
- (3) Chen. C. H. Fuzzy Logic and Neural Network  
1<sup>st</sup> Edition MacGrw-hill 1996



# Improved Genetic Algorithm for Generalized Transportation Problem

Mitsuo Gen, Kenichi Ida, Yin-Zhen Li and Juno Choi

Department of Industrial and Information System Engineering  
Ashikaga Institute of Technology  
Ashikaga 326-8558, Japan

## Abstract

In this paper, we propose a new approach of spanning tree-based genetic algorithm for Generalized Transportation Problem (GTP). Firstly, we will convert the GTP problem to a Transportation Problem (TP) because the GTP can be cast in the form in which the constraint equation and objective function resemble those of the TP model. Then, we can use spanning tree-based genetic algorithms to deal with it. In encoding method, we absorb the concept on spanning tree and adopt the Prüfer number as it is capable of equally and uniquely representing all possible basic solutions. Numerical experiments show the matrix-based genetic algorithm and spanning tree-based genetic algorithm of the proposed algorithm. **Key Words:** Genetic Algorithm, Generalized Transportation Problem, Transportation Problem

## 1 Introduction

We propose a new approach of spanning tree-based genetic algorithm for a Generalized Transportation Problem (GTP) [1]. This is true in fact of GTP that conform to the following general formulation: Given a set of  $n$  targets (e.g. production goals) and a set of  $m$  means (fixed amount of resources, instruments and so on) to fulfil those targets, and supposed each target may be fulfilled through several means and vice-versa, we are asked to *assign* the fulfillment of the various targets to the various means in such a way as to minimize a cost function. Machine-assignment problems are an outstanding example. Firstly, we will convert the GTP problems to a Transportation Problem (TP) because the GTP can be cast in the form in which the constraint equation and objective function resemble those of a TP model.

In this paper, we adopt the one-cut-point crossover operator that has been shown to be superior to traditional crossover strategies for combinatorial optimization problem. Mutation is a background operator, which produces spontaneous random changes in various chromosomes. In the evolutionary process, the mixed strategy with  $(\mu + \lambda)$ -selection [6] and roulette wheel selection [7] is used. Finally, we will show the effectiveness of the algorithm by the single objective problems. And numerical experiments show the matrix-based genetic algorithm [5] and spanning tree-based genetic algorithm of the proposed algorithm.

## 2 Problem Description

For the sake of brevity we shall write  $x_{i,n+1}$  for  $x_{is}$ . Let  $p_{ij}$  be the processing time to produce the  $j$ th product on the  $i$ th machine.  $x_{ij}$  be the amount of  $j$ th product manufactured on the  $i$ th machine.  $c_{ij}$  be the production cost of the  $j$ th product on the  $i$ th machine.  $a_i$  be the number of units available machine-hours at machine  $i$ , and  $b_j$  be the number of units demand at product  $j$ . This problem with  $m$  machines and  $n$  products can be formulated as follows:

$$\min \quad Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (1)$$

$$\text{s. t.} \quad \sum_{j=1}^n p_{ij} x_{ij} \pm x_{is} = a_i, \quad i = 1, 2, \dots, m \quad (2)$$

$$\sum_{i=1}^m x_{ij} = b_j, \quad j = 1, 2, \dots, n \quad (3)$$

$$x_{ij} \geq 0, \quad \forall i, j \quad (4)$$

We will convert the GTP problems to a TP because the GTP can be cast in the form in which the constraint equation and objective function resemble those

of a TP model.

**procedure: conversion of the GTP to TP**

**step 1.** Repeat the following substeps until  $b_j = 0, \forall j$ .

1.1 Randomly select  $i$  and  $j$ .

1.2 If  $a_i \geq b_j p_{ij}$ , then assign the available amount of units to  $x_{ij} = b_j$  and update availability  $a_i = a_i - b_j p_{ij}$  and  $b_j = b_j - x_{ij}$ . If  $a_i < b_j p_{ij}$ , then  $b_j = b_j - 1$  until  $a_i \geq b_j p_{ij}$  and assign  $x_{ij} = b_i$  and update  $a_i = a_i - x_{ij} p_{ij}$ ,  $b_j = b_j - x_{ij}$ .

**step 2.** Assign  $x_{is} = a_i, i = 1, \dots, m, s = n + 1$ .

### 3 Spanning Tree-based Genetic Algorithm

#### 3.1 Genetic Representation

Prüfer number is one of the node encoding for the tree. It can be used to encode a transportation tree. The constructive procedure for a Prüfer number according to a tree is as follows:

**procedure: convert Tree to Prüfer number**

**step 1.** Let  $i$  be the lowest numbered leaf node in tree  $T$ . Let  $j$  be the node which is the predecessor of  $i$ . Then  $j$  becomes the rightmost digit of Prüfer number  $P(T)$ ,  $P(T)$  is built up by appending digits to the right; thus,  $P(T)$  is built read from left to right.

**step 2.** Remove  $i$  and the edges  $(i, j)$  from further consideration. Thus,  $i$  is no longer considered at all and if  $i$  is the only successor of  $j$ , then  $j$  becomes a leaf node.

**step 3.** If only two nodes remain to be considered, then  $P(T)$  has been formed, and stop; else return to step 1.

Each node in the transportation problem has its quantity of supply or demand that are characterized as constraints. Therefore to construct a transportation tree, the constraint of nodes must be considered. From a Prüfer number, an unique transportation tree

also is possible to be generated by the following procedure:

**procedure: convert Prüfer number to transportation tree**

**step 1:** Let  $P(T)$  be the original Prüfer number and let  $\bar{P}(T)$  be the set of all nodes that are not part of  $P(T)$  and design as eligible for consideration.

**step 2:** Repeat the process 2.1-2.4 until no digits left in  $P(T)$ .

2.1 Let  $i$  be the lowest numbered eligible node in  $\bar{P}(T)$ . Let  $j$  be the leftmost digit of  $P(T)$ .

2.2 If  $i$  and  $j$  not in the same set  $O$  or  $D$ , then add the edge  $(i, j)$  to tree  $T$ . Otherwise, select the next digit  $k$  from  $P(T)$  that not included in the same set with  $i$ , exchange  $j$  with  $k$  and add the edge  $(i, j)$  to the tree  $T$ .

2.3 Remove  $j$ (or  $k$ ) from  $P(T)$  and  $i$  from  $\bar{P}(T)$ . If  $j$ (or  $k$ ) does not occur anywhere in the remaining part of  $P(T)$ , then put it into  $\bar{P}(T)$ . Designate  $i$  as no longer eligible.

2.4 If  $a_i \geq b_j p_{ij}$ , then assign the available amount of units to  $x_{ij} = b_j$  and update availability  $a_i = a_i - b_j p_{ij}$  and  $b_j = b_j - x_{ij}$ . If  $a_i < b_j p_{ij}$ , then  $b_j = b_j - 1$  until  $a_i \geq b_j p_{ij}$  and assign  $x_{ij} = b_i$  and update  $a_i = a_i - x_{ij} p_{ij}$  and  $b_j = b_j - x_{ij}$ .

**step 3:** If no digits remain in  $P(T)$ , then there are exactly two nodes,  $i$  and  $j$  still eligible in  $\bar{P}(T)$  for consideration. Add edge  $(i, j)$  to tree  $T$  and form a tree with  $m + n - 1$  edges.

**step 4:** If no available amount of units to assign, then stop. Otherwise, there are remaining supply  $r$  and demand  $s$ , add edges  $(r, s)$  to tree and assign the available amount of units  $x_{rs} = a_r = b_s$  to edges. If there exists a cycle, then remove the edge that assigned zero flow. A new spanning tree is formed with  $m + n - 1$  edges.

#### 3.2 Initialization

The initialization of a chromosome(a Prüfer number) is performed from that randomly generated  $m + n - 2$

digits in range  $[1, m+n]$ . However, it is possible to generate an infeasible chromosome which is not adapted to generate a transportation tree.

We design the handling for feasibility of the chromosome with the following criterion: Denote that  $S_O$  and  $S_D$  are the sum of connections of nodes which are included in set  $O$  and  $D$  respectively from  $P(T)$ . Also we denote that  $\bar{S}_O$  and  $\bar{S}_D$  are the appearing times of those nodes in  $\bar{P}(T)$  and included in set  $O$  and  $D$ , respectively. If  $S_O + \bar{S}_O = S_D + \bar{S}_D$ , then  $P(T)$  is feasibility. Otherwise  $P(T)$  is infeasible.

### 3.3 Genetic Operators

Here, the one-cut-point crossover operator is used. For avoiding unnecessary decoding from which an infeasible chromosome (a Prüfer number) may be generated after crossover operator, we add the criterion for feasibility of the chromosome. A Prüfer number via this criterion is always feasible and can be decoded into a corresponding transportation tree.

The inversion mutation and displacement mutation are used. This two mutation operators always generate feasible chromosomes if the parents are feasible, because the criterion  $S_O + \bar{S}_O = S_D + \bar{S}_D$  is unchanged after these operators.

### 3.4 Evaluation and Selection

In this approach, the evaluation procedure consists of the following two steps: (1) convert a chromosome into a tree, and (2) calculate a objective function. It is given as follows:

**procedure: evaluation**

**begin**

$T \leftarrow \{\phi\}; p \leftarrow 0;$

define  $\bar{P}(T)$  according to the  $P(T)$ ;

**repeat**

select the leftmost digit from  $P(T)$ , say  $i$ ;

select the eligible node with the lowest

numbered from  $\bar{P}(T)$ , say  $j$ ;

**if**  $i, j \in O$  or  $i, j \in D$  **then**

select next digit from  $P(T)$  not in the same

set with  $j$ , say  $k$ ;

exchange  $k$  with  $i$  and take  $i \leftarrow k$ ;

**endif**

**if**  $a_i \geq b_j p_{ij}$  **then**

assign the flow:

$x_{ij} \leftarrow b_j; T \leftarrow T \cup x_{ij};$

$z(T) \leftarrow z(T) + c_{ij} x_{ij};$

update available amount:

$a_i \leftarrow a_i - b_j p_{ij}; b_j \leftarrow b_j - x_{ij};$

**endif**

**if**  $a_i < b_j p_{ij}$  **then**

**repeat**

$b_j \leftarrow b_j - 1$

**until**  $(a_i \geq b_j p_{ij});$

assign the flow:

$x_{ij} \leftarrow b_j; T \leftarrow T \cup x_{ij};$

$z(T) \leftarrow z(T) + c_{ij} x_{ij};$

update available amount:

$a_i \leftarrow a_i - b_j p_{ij}; b_j \leftarrow b_j - x_{ij};$

**endif**

remove  $i$  from  $P(T)$ ;

remove  $j$  from  $\bar{P}(T)$ ;

**if**  $i$  does not occur anywhere in remaining

$P(T)$  **then** put  $i$  into  $\bar{P}(T)$ ;

**endif**

$p \leftarrow p + 1;$

**until**  $(p \leq m + n - 2)$

assign the flow for  $r, s \in \bar{P}(T)$ :

$x_{rs} \leftarrow b_j, r \in O, s \in D;$

$T \leftarrow T \cup x_{ij};$

$z(T) \leftarrow z(T) + c_{ij} x_{ij};$

**if** there exists a cycle **then**

find edges with zero flow and remove it

**endif**

**end**

In the selection procedure, we will use the mixed strategy with  $(\mu + \lambda)$ -selection and roulette wheel selection can enforce the best chromosomes into the next generation. This strategy selects  $\mu$  parents and  $\lambda$  offspring. If there are no  $\mu$  different chromosomes available, then the vacant pool of population is filled up with roulette wheel selection.

## 4 Numerical Experiments

Let us consider the generalized transportation problem with the data given in Table 1. The figures in the left upper corner of each cell of columns 1,  $\dots$ , 7, respectively  $s(8)$ , are the  $p_{ij}$ , respectively the coefficients of

$x_{is}$ . The figures in the right lower corner of each cell are the  $c_{ij}$ . The last column contains the  $a_i$ , while the last row the  $b_j$ .

Table 1: Data of the generalized transportation problem

	1	2	3	4	5	6	7	8	$a_i$
1	3	2	3	4	5	3	2	1	120
2	8	26	13	20	15	16	16	1	130
3	14	17	23	24	9	18	20	1	140
4	26	15	16	18	12	15	8	1	145
5	2	4	5	4	3	2	5	1	150
6	9	12	23	15	16	9	4	1	145
7	3	5	6	4	3	6	6	1	150
8	17	18	16	40	16	28	16	1	145
9	4	5	6	3	4	5	7	1	145
$b_j$	6	5	9	7	8	5	6		

Table 2: Average results by the matrix-based GA and tree-based GA approach GTP

problem size $m \times n$	parameter		matrix-based GA		tree-based GA	
	pop-size	max-gen	percent	ACT(s)	percent	ACT(s)
3 × 4	20	200	100	0.95	100	0.35
4 × 5	20	500	100	5.39	100	1.25
5 × 6	50	500	100	18.46	100	5.35
6 × 7	100	1000	100	2038.10	100	801.27

percent: the percentage of running times that obtained optimal solution

ACT: the average computing time

The simulations were carried to out on each example with their best mechanism (parameter setting). Table 2 shows average simulation results from 20 times running by the spanning tree-based genetic algorithm (st-GA) and matrix-based genetic algorithm (m-GA). From Table 2 we know that the optimal solution was found accurately by the st-GA with more short time than m-GA in each running time. Figure 1 shows the evolutionary processes with m-GA and st-GA on a generalized transportation problem with 5 machines and 6 products. This figure clearly shows that the spanning tree-based permutation has better mechanism to evolve to the optimal solution in the evolutionary process than the matrix-based representation.

## 5 Conclusion

In this paper, we proposed a new approach by using spanning tree-based genetic algorithm to solve the generalized transportation problem. Spanning tree-based encoding was implemented with Prüfer number. We carried out the numerical experiments by both the m-GA and the st-GA. For the small scale problem, there is no great difference on the computing results, which shows the effectiveness of the proposed GA approach. On the larger scale problem, the proposed GA approach can get the optimal solution with more

less CPU time than the m-GA approach.

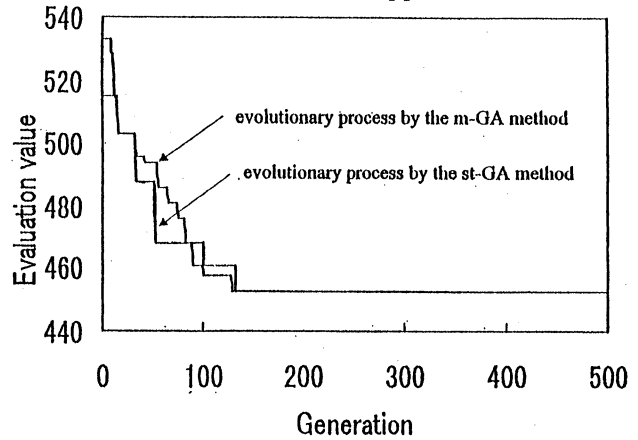


Figure 1: Evolutionary process with m-GA and st-GA on the GTP

## References

- [1] Balas, E. and Ivanescu, P.L., "On the Generalized Transportation Problem," *Management Science*, vol. 11, No. 1, 1964, pp. 188-202.
- [2] Charnes, A. and Cooper, W. W. "Management Models and Industrial Applications of Linear Programming", *Management Science*, Vol.4, No. 1, 1957.
- [3] Gen, M. and Y.Z. Li, Solving Multiobjective Solid Transportation Problem by Genetic Algorithm, *Journal of Japan Industrial Management Assoc.*, pp.446-454, 1995 (in Japanese)
- [4] Li, Y.Z. and M. Gen, "Spanning Tree-Based Genetic Algorithm for Bicriteria Transportation Problem", *Proc. of The Australia-Japan Joint Workshop on Intelligent and Evolutionary Systems*, p.112-119, Canberra, 1997.
- [5] Michalewicz, Z., *Genetic Algorithms + Data structure = Evolution Programs*, Springer-Verlag, 1992; Second, extended ed., 1994; Third revised and extended ed., 1996.
- [6] Schwefel, H., *Evolution and Optimum Seeking*, John Wiley and Sons, New York, 1994.
- [7] Zhou, G. and M. Gen, Approach to Degree-Constrained Minimum Spanning Tree Problem Using Genetic Algorithm, *Engineering Design and Automation*, Vol 3, No.2, pp.157-165, 1997.

# Genetic Algorithms Approach on Leaf-Constrained Spanning Tree Problem

Gengui Zhou Mitsuo Gen

Department of Industrial and Information Systems Engineering  
Ashikaga Institute of Technology, Ashikaga 326-8558, Japan

## Abstract

In this paper we discuss the terminal layout problem with the constraint limiting the number of terminals with "degree 1" in a tree solution. In graph theory terminology, the problem can be stated as the Leaf-Constrained Minimum Spanning Tree (LMST) problem. Because of its NP-hard complexity, there are still no effective algorithms to solve this problem. We present a new approach by using the Genetic Algorithms (GAs) to deal with the LMST problem. Compared with the lower and upper bounds obtained by a simple method, Numerical results shows the effectiveness of the proposed GAs approach on the LMST problem.

Keywords: Genetic Algorithms, Spanning Tree, Leaf-Constraint.

## 1 Introduction

In the context of telecommunication networks, the network terminals involve certain constraints which are either related with the performance of the corresponding network or with the availability of some classes of devices. One of them is the terminal layout problem with the constraints limiting the number of terminals with "degree 1" in the solution. Here, by the degree is meant the number of wires (edges) incident to the terminals (nodes). This type of constraints more often appear when designing tree-like networks.

In the graph theory terminology, the term "leaf" is used for denoting an edge which is incident to a node of degree 1. Therefore, the above constraint can be restated as "the tree solution has to contain exactly  $k$  leaves" or the Leaf-Constrained Minimum Spanning Tree (LMST) problem<sup>1</sup>. The LMST is NP-hard because it contains as a particular case, the case  $k = 1$ , which is the shortest Hamiltonian path problem between a given node and any other node. This problem is equivalent to the well-known traveling salesman problem and hence, the LMST is also NP-hard. As far as we know, no efficient algorithm for this problem has been given in literature.

In this paper, we present a new approach by using the genetic algorithms (GAs), which have been demonstrating their powerful potential in dealing with such complicated combinatorial problem with tree topology<sup>2-7</sup>. For the adaptation to the evolutionary

process, we developed a tree-based genetic representation to code the candidate solutions of the LMST problem. In order to illustrate the effectiveness of the proposed GAs approach, we simply design a strategy to give out the lower and upper bounds of the problem. Compared with its lower and upper bounds, the results on numerical experiments clearly show the effectiveness of the proposed GAs approach on the LMST problem.

## 2 Model for the LMST

### 2.1 Problem Description

In this section we formulate the LMST problem as a zero-one integer programming. Considering a complete, undirected graph  $G = (V, E)$ , we let  $V = \{1, 2, \dots, n\}$  be the set of nodes representing the terminals and the central site, or "root" node, as node 1, and  $E = \{(i, j) | i, j \in V\}$  be the set of edges representing all possible telecommunication wiring. For a subset of nodes  $S (\subseteq V)$  we define  $E(S) = \{(i, j) | i, j \in S\}$  to be the edges whose end points are both in  $S$ . Also, we define the following binary decision variables:  $x_{ij} = 1$ , if edge  $(i, j)$  is selected in the tree, otherwise,  $x_{ij} = 0$  for all edges  $(i, j) \in E$ .

To formulate the LMST we add the binary variables  $U_j, j = 1, 2, \dots, n$ , such that  $U_j = 1$  if node  $j$  has degree 1 in the solution and  $U_j = 0$ , otherwise. The constraint on the number of leaves can now be written as:  $\sum_{j=1}^n U_j = k$

Clearly, we need to add the following set of constraints which link the  $x_{ij}$  variables with the  $U_j$  variables.

$$1 - U_i \leq \sum_{j=1}^n x_{ij} \leq (n-1)(1 - U_i), i = 1, 2, \dots, n$$

These constraints guarantee that node  $i$  has degree 1 ( $U_i = 1$ ) if and only if no arcs are leaving that node. Notice also that the right-hand inequality of above equations can be tightened to

$$1 - U_i \leq \sum_{j=1}^n x_{ij} \leq k(1 - U_i), i = 1, 2, \dots, n$$

because the outdegree of any node cannot be greater than  $k$ . Finally, the  $\{0, 1\}$  requirements for the new

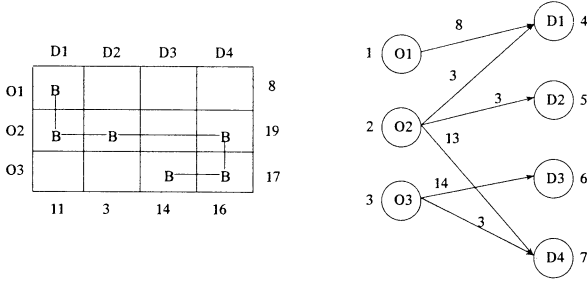


Figure 1: A spanning tree and its Prüfer number

of the set  $S = \{1, 2, \dots, m\}$  and denote that consumers  $m + 1, \dots, m + n$  as the component of the set  $D = \{m + 1, \dots, m + n\}$ .

### Representation

Transportation graph in Figure 1 can be represented as a spanning tree with Prüfer number such as in Figure 2.

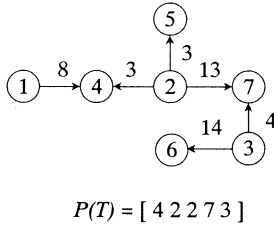


Figure 2: A spanning tree and its Prüfer number

For a complete graph with  $p$  nodes, there are  $p^{(p-2)}$  distinct labeled trees. This means that we can use only permutation of  $p - 2$  digits in order to uniquely represent a tree with  $p$  nodes where each digit is an integer between 1 and  $p$  inclusive. For any tree there are always at least two leaf nodes.

Prüfer number is one of the node encoding for the tree. It can be used to encode a transportation tree. Each node in the fcTP has its quantity of supply or demand, which are characterized as constraints. Therefore to construct a transportation tree, the constraint of nodes must be considered. From a Prüfer number, an unique transportation tree also is possible to be generated.

In Figure 2, the Prüfer number  $[4 2 2 7 3]$  corresponds to a spanning tree on a 7-node complete graph. The construction of the Prüfer number is described as follows: Locate the leaf node having the lowest number. In this case, it is node 1. Since node 4 is incident to node 1 in the tree, assign 4 to the first digit in the permutation, then remove node 4 and edge (1,4). Now, node 4 is the lowest numbered leaf node and node 2 is incident to it, assign 2 to the second digit in the permutation and then remove node 2 and edge (4,2).

Repeat the process on the subtree until edge (3,7) is left and the Prüfer number of this tree with 7 digits is finally produced.

Conversely, the corresponding tree using the Prüfer number can be constructed. As the node 1, 5 and 6 are not included in  $P(T)$ , let  $\bar{P}(T)$  be the set of them as  $\bar{P}(T) = \{1, 5, 6\}$ . Node 1 is the lowest number in  $\bar{P}(T)$ . Node 4 is left most digit of  $P(T)$ . Check this two digit 1 and 4, they are not in the same set that  $1 \in S$ ,  $4 \in D$ . Add edge (1,4) to the tree, remove node 1 from  $\bar{P}(T)$  and leftmost digit 4 of  $P(T)$  leaving  $P(T) = [2 2 7 3]$ . Because node 4 does not occur in the remaining part of  $P(T)$ , put it into  $\bar{P}(T) = \{4, 5, 6\}$ . Assign the available amount of units to  $x_{14} = \min\{a_1, b_4\} = \min\{8, 11\} = 8$ . Update  $a_1 = 8 - x_{14} = 0$  and  $b_4 = 11 - x_{14} = 3$ .

Secondly, node 4 is now the lowest digit in  $\bar{P}(T)$  and node 2 is the leftmost digit in remaining  $P(T)$ . Since  $2 \in S$  and  $4 \in D$ , add edge (2,4) to the tree, remove 4 from  $\bar{P}(T)$  and the leftmost digit 2 from  $P(T)$  leaving  $P(T) = [2 7 3]$ . Because node 2 occur in the remaining part of  $P(T)$ , does not put it into  $\bar{P}(T) = \{5, 6\}$ . Assign the available amount of units to  $x_{24} = \min\{19, 3\} = 3$ . Update  $a_2 = 19 - x_{24} = 16$  and  $b_4 = 3 - x_{24} = 0$ .

Repeat the process until  $P(T) = [3]$  and nodes 6 and 7 are in  $\bar{P}(T)$ . Add edge (3,6) to the tree, remove the last digit 3 of  $P(T)$  and node 6 from  $\bar{P}(T)$ . As node 3 no longer occurs in the remaining  $P(T)$ , it needs putting into  $\bar{P}(T)$  leaving  $\bar{P}(T) = \{3, 7\}$ . Finally,  $P(T)$  is empty and only nodes 3 and 7 are in  $\bar{P}(T)$ . Add edge (3,7) to the tree and stop. The tree in Figure 2 is formed.

### Initialization and Feasibility of the Chromosome

The initialization of a chromosome (a Prüfer number) is performed from that randomly generated  $m + n - 2$  digits in range  $[1, m + n]$ . However, it is possible to generate an infeasible chromosome, which is not adapted to generate a transportation tree.

Prüfer number encoding contains nodes and connections information. From this encoding we can easily to create feasibility of chromosomes.

Denote that  $L_S$  and  $L_D$  are the sum of connections of nodes which are included in set  $S$  and  $D$  respectively from  $P(T)$ . Also we denote that  $\bar{L}_S$  and  $\bar{L}_D$  are the appearing times of those nodes in  $\bar{P}(T)$  and included in set  $S$  and  $D$ , respectively. We design the handling for feasibility of the chromosome with the following criterion:

**Feasibility of Chromosome:** If  $L_S + \bar{L}_S = L_D + \bar{L}_D$ , then  $P(T)$  is feasibility. Otherwise infeasible.

### Genetic Operators

**Crossover:** the one-cut-point crossover operation is used.

**Mutation:** the inversion mutation and displacement mutation are used. The inversion mutation is selects two positions within a chromosome at random and then inverts the substring between these two positions.

This two mutation operators always generate feasible chromosomes if the parents are feasible, because the criterion  $L_S + \bar{L}_S = L_D + \bar{L}_D$  is unchanged after these operations.

### Evaluation and Fitness Function

In this approach, The evaluation procedure also consists of two steps:

- (1) convert a chromosome into a tree, and
- (2) calculate each objective function.

In multi-criteria optimization context, usually the Pareto optimal solutions are characterized as the solutions of the multi-objective programming problem. In this stage, the module for Pareto optimal solutions consists of two steps: (1) evaluate chromosomes by the objective functions, and (2) select Pareto solutions based on evaluation values. Let  $E(t)$  be the Pareto solution set generated up to current iteration  $t$ , then the procedure for Pareto solutions is given as follows:

### Procedure: Pareto solutions

**Step 1:** Set iteration  $t = 0$ , and  $E(t) = \{\emptyset\}$ . Let  $t_m$  be the maximum iteration.

**Step 2:** If  $t > t_m$  then Stop. Otherwise go to Step 3.

**Step 3:** Set the counter  $k = 1$ , and count the number of current chromosomes  $i\_size$ .

**Step 4:** Repeat Step (4.1)—(4.3) to update  $E(t)$  until  $k \leq i\_size$ .

(4.1) Evaluate a chromosome  $T_k$  and obtain the solution vector  $\mathbf{z}_k = [z_1(T_k) \ z_2(T_k)]$ .

(4.2) Register Pareto solutions into  $E(t)$  and delete non-Pareto solutions in  $E(t)$  by the solution vector  $\mathbf{z}_k$ .

(4.3) Set  $k = k + 1$ .

**Step 5:** Set  $t = t + 1$ , go to Step 2.

**Fitness function:** the weighted sums method is used to construct the fitness function, which the bicriteria functions dynamically are combined into one overall objective function at hand [4].

### Selection

In the selection procedure, we will use the mixed strategy with  $(\mu + \lambda)$ -selection and roulette wheel selection can enforce the best chromosomes into the next generation.

## 4 Numerical Experiments

The genetic algorithm called st-GA was implemented in C language and run on a HP 9000 Model 715/100 workstation. We have simulated using four bicriteria fixed charge transportation problems. The computational experiments with five problem sizes are used, denoted by  $m \times n$ , those are  $4 \times 5$ ,  $5 \times 10$ ,  $8 \times 10$ ,  $10 \times 15$ , and  $15 \times 20$ . The total supply (demand) for different problem size are given Table 1. The unit variable costs and unit shipping times in all test problems are integers ranging from 3 through 8 and from 1 to 7, respectively. The fixed costs for all test problems are generated randomly as in Table 1.

Table 1: Total supply (demand) and ranges of fixed cost for different problem sizes

Problem size	Total supply	Fixed cost
$4 \times 5$	3000	[50, 100]
$5 \times 10$	5000	[100, 500]
$8 \times 10$	10000	[500, 800]
$10 \times 15$	15000	[700, 1000]

The simulations were carried out on each problem with their best mechanism (parameter setting), and run by 10 times. Table 4 given the data of  $4 \times 5$  problem. Figure 3 shown the summarized solutions of obtained in 10 trials by st-GA for bicriteria problem  $4 \times 5$  with given parameters mutation rate 0.4 and crossover rate 0.2. The ideal solution for  $4 \times 5$  problem has solved (10809, 8250) which denoted by \* notation in Figure 3. From Figure 3, we know the solutions were obtained Pareto frontier which contain a extreme solution point. In some trials, the algorithm got the different solutions in the criteria space. This is because the fitness function was dynamically created with two objective functions in the evolutionary process, therefore, the mechanism of the st-GA in each trial with different search direction to evolve the solutions. The average computing time of each case is summarized in Table 3.

## 5 Conclusion

In order to improve the efficiency of genetic algorithm to solve bicriteria fixed charge transportation problem, the spanning tree was used to represent the data structure of the problem. In the evolutionary process the fitness function based on two objective functions was created dynamically and it has to evolve the problem solutions. Also we compared with the m-GA and st-GA by four numerical experiments. The proposed st-GA is more saving than m-GA concerning the computation time on the bicriteria fixed charge transportation problem.

Table 2: Generated data for problem  $4 \times 5$ 

$i$	Transportation cost					Transportation time					Fixed costs					Supply
1	8	4	3	5	8	3	3	3	5	3	60	88	95	76	97	857
2	3	6	4	8	5	4	2	1	5	4	51	72	65	87	76	893
3	8	4	5	3	4	2	1	5	3	2	67	89	99	89	100	505
4	4	6	8	3	3	2	2	4	3	4	86	84	70	92	88	745

Demand:  $b_1=888$   $b_2=157$   $b_3=284$   $b_4=773$   $b_5=898$

Table 3: Comparison with m-GA and st-GA on bicriteria fcTP

$m$	$n$	Memory for a solution		ACT(s)		Parameter	
		m-GA	st-GA	m-GA	st-GA	$pop\_size$	$max\_gen$
4	5	20	7	18.78	10.54	50	1000
5	10	50	13	71.27	23.75	100	1000
8	10	80	16	6802.36	986.66	100	2000
10	15	150	23	12145.82	11237.75	200	2000

ACT: average computing time

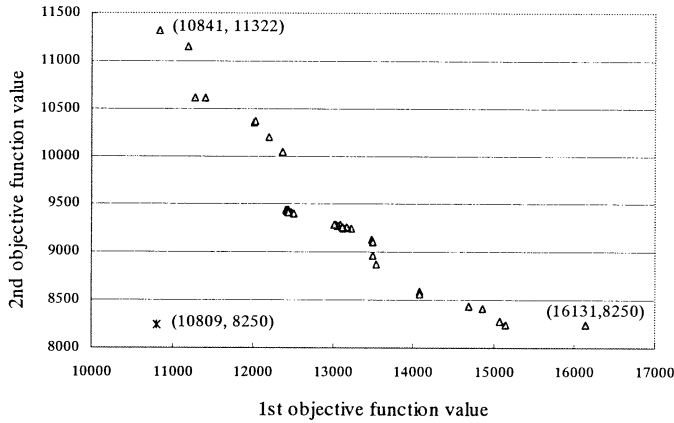


Figure 3: Obtained solutions by st-GA

## References

- [1] Gen, M. and R. Cheng, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, New York, 1997.
- [2] Gen, M. and Y.Z. Li, Solving Multi-objective Transportation Problem by Spanning Tree-based Genetic Algorithm, (I. Parmee, ed.) *Adaptive Computing in Design and Manufacture*, pp.95–108, Springer-Verlag, 1998.
- [3] Gottlieb, J. and L. Paulmann, Genetic Algorithms for the Fixed Charge Transportation Problem, *Proc. of IEEE International Conference on Evolutionary Computation*, pp.330–335, Anchorage, 1998.
- [4] Hirsch, W.M. and G.B. Dantzig, The Fixed Charge Problem, *Naval Research Logistics Quarterly* 15, pp.413–424, 1968.
- [5] Li, Y.Z. and M. Gen, "Spanning Tree-Based Genetic Algorithm for Solving Bicriteria Transportation Problem", *Journal of Japan Society for Fuzzy Theory and Systems*, Vol.10, No.5, pp.888–898, 1998.
- [6] Li, Y.Z. and M. Gen, "Fixed Charge Transportation Problem by Spanning Tree-Based Genetic Algorithm", *Beijing Mathematics*, Vol.4, No.2, pp.239–249, October 1998.
- [7] Michalewicz, Z., G. A. Vignaux and M. Hobbs, A Non-Standard Genetic Algorithm for the Nonlinear Transportation Problem, *ORSA Journal on Computing*, Vol.3, No.4, pp.307–316, 1991.
- [8] Salkin, H. M. and K. Mathur, *Foundations of Integer Programming*, North-Holland, 1989.
- [9] Steinberg, D.I., The Fixed Charge Problem, *Naval Research Logistics Quarterly*, Vol. 17, No.2, pp.217–236, 1970.
- [10] Sun, M., J.E. Aronson, P.G. Mckeown, and D. Drinka, A Tabu Search Heuristic Procedure for the Fixed Charge Transportation Problem, *European J. of Operational Research* 106, pp.441–456, 1998.
- [11] Taha, H.A., *Integer Programming*, Academic Press, 1975.
- [12] Zhou, G. and M. Gen, A Note on Genetic Algorithms for Degree-Constrained Spanning Tree Problem, *Networks*, Vol.30, No.2, pp.91–96, 1997.



## A Genetic Algorithm for Bicriteria Fixed Charge Transportation Problem

Yinzhen Li, Mitsuo Gen and Kenichi Ida

Dept. of Industrial and Information Systems Engg.,  
Graduate School of Engineering,  
Ashikaga Institute of Technology,  
Ashikaga, 326-8558 Japan

### Abstract

In this paper, we presented a genetic algorithm for solving bicriteria fixed charge transportation problem. The genetic algorithm is created with spanning tree representation. Due to the bicriteria program, the fitness is evaluated based on overall objective function constructed by weighted sum method. The genetic algorithm can found the Pareto optimal solutions in the bicriteria space. Computational results will show the performance of the genetic algorithm.

### 1 Introduction

The fixed charge transportation problem (fcTP) is a more practical transportation problem (TP). Many practical transportation and distribution problems such as the minimum cost network flow (transshipment) problem with fixed charge in logistics can be formulated as fixed charge transportation problems. Often the representation of more than one objectives are desired expect for transportation cost in the models of economic decisions in the production-distribution problems. For example, delivery time of the commodities, reliability of transportation and so on. The bicriteria-programming problem is a special case of multiobjective programming problem. Many solution procedures have been investigated for bicriteria program and extended to the multiobjective programming problems.

Recently, for solving the fcTP, Gottlieb *et. al.* [2] proposed a genetic algorithm (GA) based on permutation representation, and a tabu search heuristic procedure was proposed by Sun *et. al.* [9]. Li *et. al.* proposed a spanning tree-based GA for solving fcTP, and shown the comparison with matrix-based GA [5] and given the conclusion that spanning tree-based GA saving more computational time than matrix-based GA and has higher solution quality.

In this paper, we extend the spanning tree-based GA to the bicriteria fixed charge transportation problem. The mechanism of the GA absorbs the spanning tree structure and foundation of the genetic algorithm. In the evolutionary process, the fitness function dynamically is created by the weighted sum method. The mixed strategy for genetic selection is used in evolutionary process. Finally, the numerical experiments

will show the performance of the GA.

### 2 Bicriteria Fixed Charge Transportation Problem

The Bicriteria Fixed Charge transportation problem (b-fcTP) with given  $m$  plants and  $n$  consumers can be formulated as follows:

$$\begin{aligned} \min \quad & f_1(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n (f_{ij}(\mathbf{x}) + d_{ij}g_{ij}(\mathbf{x})) \\ \min \quad & f_2(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n t_{ij}(\mathbf{x}) \\ \text{s. t.} \quad & \sum_{j=1}^n x_{ij} \leq a_i, \quad i = 1, 2, \dots, m \\ & \sum_{i=1}^m x_{ij} \geq b_j, \quad j = 1, 2, \dots, n \\ & x_{ij} \geq 0, \quad \forall i, j \\ \text{with} \quad & g_{ij}(\mathbf{x}) = \begin{cases} 1, & \text{if } x_{ij} > 0 \\ 0, & \text{otherwise} \end{cases} \end{aligned}$$

where  $\mathbf{x} = [x_{ij}]$  is the unknown quantity to be transported on the route  $(i, j)$  that from plant  $i$  to consumer  $j$ .  $f_{ij}(\mathbf{x})$  can be viewed as the objective function from original TP that means total transportation cost in which  $f_{ij}(\mathbf{x}) = c_{ij}x_{ij}$  will be a cost function for shipping per unit from plant  $i$  to consumer  $j$  if it is linear.  $d_{ij}$  is the fixed cost associated with route  $(i, j)$ , and  $t_{ij}(\mathbf{x})$  can be viewed as a function of delivery time in which in the form  $t_{ij}(\mathbf{x}) = t_{ij}x_{ij}$  where  $t_{ij}$  is per unit delivery time from plant  $i$  to consumer  $j$ .  $a_i$  is the number of units available at plant  $i$ , and  $b_j$  is the number of units demanded at consumer  $j$ .

### 3 Genetic Algorithm

This genetic algorithm was completed with spanning tree representation. As a matter of convenience we denote that plants  $1, 2, \dots, m$  as the component

variables have to be included in the model. Then, the LMST problem can be formulated as the following integer programming:

$$\min z = \sum_{i=1}^{n-1} \sum_{j=2}^n c_{ij} x_{ij} \quad (1)$$

$$\text{s. t. } \sum_{i=1}^{n-1} \sum_{j=2}^n x_{ij} = n - 1 \quad (2)$$

$$\sum_{i \in S} \sum_{\substack{j \in S \\ j > 1}} x_{ij} \leq |S| - 1, S \subseteq V \setminus \{1\}, |S| \geq 2 \quad (3)$$

$$1 - u_i \leq \sum_{j=1}^n x_{ij} \leq k(1 - u_i), i = 1, 2, \dots, n \quad (4)$$

$$x_{ij} = 0 \text{ or } 1, i = 1, \dots, n-1, j = 2, \dots, n \quad (5)$$

$$u_j = 0 \text{ or } 1, j = 1, 2, \dots, n \quad (6)$$

where  $c_{ij}$  be the (fixed) cost of including edge  $(i, j)$  in the solution

## 2.2 Upper and Lower Bounds

Simply, we can obtain a MST solution after applying the Prim's algorithm<sup>8</sup>. If the number of all leaves are equal to the required leaf-constrained value, this solution is definitely the optimal solution of the problem. But, it is not always true in most cases. If we do not consider the constraints of Eq.4, Eq.5 and Eq.6, obviously, this MST solution of the problem can be regarded as its lower bound.

On the other hand, usually the number of leaves in a MST solution is more than the required leaf-constrained value (we only consider those cases that the required leaf-constrained value is much less than the total nodes, otherwise, there may be no meaning in limiting the number of leaves). In this case, if the MST solution is modified by decreasing its number of leaves, we can easily obtain an upper bound of the problem. The operation can be described as follows:

### procedure: upper bounds calculation

**step 1:** Determine all one-leaf subtrees  $P_1, P_2, \dots, P_q$ .

**step 2:** If  $q = k$ , the current MST solution is the LMST solution and stop; otherwise, goto **step 3**.

**step 3:** Calculate the cost if  $P_l$  is appended on other one-leaf subtrees. Suppose that  $w_{lr}$  is the minimum cost for  $P_l$  being appended on  $P_r$  ( $1 \leq r \leq q, r \neq l$ ).

**step 4:** Repeat **step 3** for  $l = 1, 2, \dots, q$  and calculate the minimum cost  $w_{st}$ , where  $w_{st} = \min\{w_{lr} \mid l, r = 1, 2, \dots, q, l \neq r\}$ .

**step 5:** Append one-leaf subtree  $P_s$  to one-leaf subtree  $P_t$ , set  $q \leftarrow q - 1$  and goto **step 1**.

## 3 Genetic Algorithm Approach

### 3.1 Genetic Representation

As to the MST problem, two main factors should be taken into consideration if we want to keep its tree

topology in the genetic representation: one is the connectivity among nodes; the other is the degree value (the number of edges connected on it) of each node. Therefore, the intuitive idea of encoding a tree solution is to use a two-dimension structure for its genetic representation<sup>7</sup>. One dimension encodes the nodes of a spanning tree; another dimension encodes the degree value of each node. Based on this observation, the tree-based permutation of such a tree can be encoded as the following procedure:

### procedure: tree-based encoding

**step 1:** Select node 1 (root node) in a labeled tree  $T$ , put it as the first digit in the node dimension of the permutation and its degree value as the first digit in the degree dimension, and determine that node as the current node.

**step 2:** Check the successor node of the current node from left branch to right branch. If there is a successor node, put it to the permutation in the node dimension and its degree value to the permutation in the degree dimension (here we build the permutation by appending digits to the right), and then goto **step 3**. If there is no such node, determine the predecessor node as the current node, and return to **step 2**.

**step 3:** If the successor node is not a leaf node, determine the successor node as the current node, then goto **step 2**. If the successor node turns to be a leaf node, delete it and goto **step 4**.

**step 4:** If all nodes have been checked, stop; otherwise, goto **step 2**.

Figure 1 illustrates an example of this tree-based permutation.

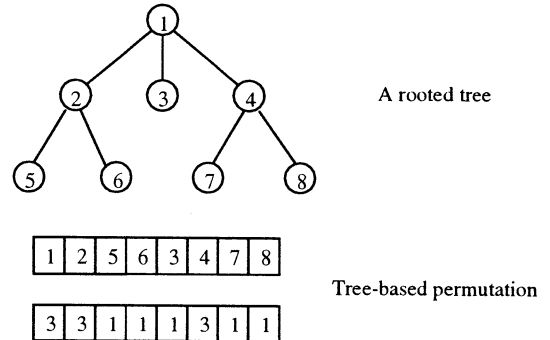


Figure 1: A rooted tree and its tree-based permutation

Also, it is easy to decode the above tree-based permutation into a tree. Supposed that the vertex dimension for individual  $P$  is represented as  $P_1(k), k = 1, 2, \dots, n$  and the degree dimension for individual  $P$  as  $P_2(k), k = 1, 2, \dots, n$ . The decoding procedure for each individual in the form of degree-based permutation can be operated as follows:

### procedure: degree-based permutation decoding

**step 1:** Set  $k \leftarrow 1$  and  $j \leftarrow 2$ .

**step 2:** Select the vertex in  $P_1(k)$ , say  $v_r$  where  $r = P_1(k)$ , select the vertex in  $P_1(j)$ , say  $v_s$  where  $s = P_1(j)$ , add the first edge from  $r$  to  $s$  into a tree.

**step 3:** Let  $P_2(j) \leftarrow P_2(j) - 1$ .  
**step 4:** If  $P_2(j) \geq 1$ , let  $k \leftarrow j$  and  $j \leftarrow j + 1$ .  
**step 5:** If  $k < 1$ , stop.  
**step 6:** If  $P_2(k) \geq 1$ , select the vertex in  $P_1(k)$ , say  $v_r$ , where  $r = P_1(k)$  and let  $P_2(k) \leftarrow P_2(k) - 1$ ; otherwise, let  $k \leftarrow k - 1$  and go to **step 5**.  
**step 7:** If  $P_2(j) \geq 1$ , select the vertex in  $P_1(j)$ , say  $v_s$ , where  $s = P_2(j)$ , and let  $P_2(j) \leftarrow P_2(j) - 1$ ; otherwise, let  $j \leftarrow j + 1$  and go to **step 7**.  
**step 8:** Add the edge from  $r$  to  $s$  into the tree and go to **step 4**.

### 3.2 Genetic Operation

**Order Crossover:** The crossover is only operated on node dimension, not together with the degree dimension. The operation can be illustrated by Figure 2. It is not difficult to image that the crossover will dramatically change the tree structure among generations. But, it is useful for the exploration in the evolutionary process.

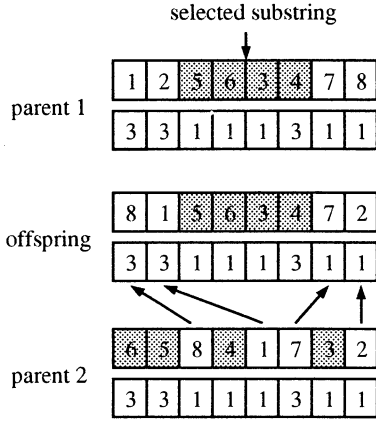


Figure 2: Order crossover on nodes

**Exchange Mutation:** Exchange mutation selects two genes (nodes) at random and then swaps the genes (nodes). This mutation is essentially a 2-opt exchange heuristic. The operation can be illustrated by Figure 3.

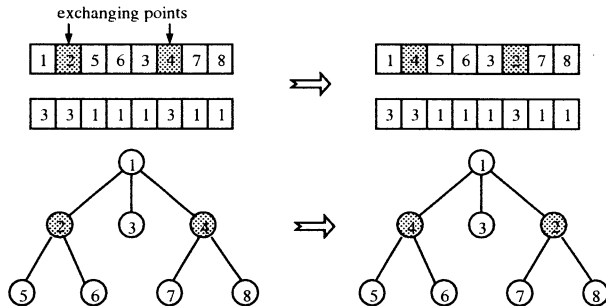


Figure 3: Exchange mutation on nodes

**Insertion Mutation:** Insertion mutation selects a string of genes (branch) at random and inserts it in a random gene (node). The operation can be illustrated by Figure 4. Obviously, this operation is indispensable

for the evolutionary process to evolve to a highly fit tree structure. Because it may keep good heritages from parents among generations. As a result, it makes the exploitation in the evolutionary process.

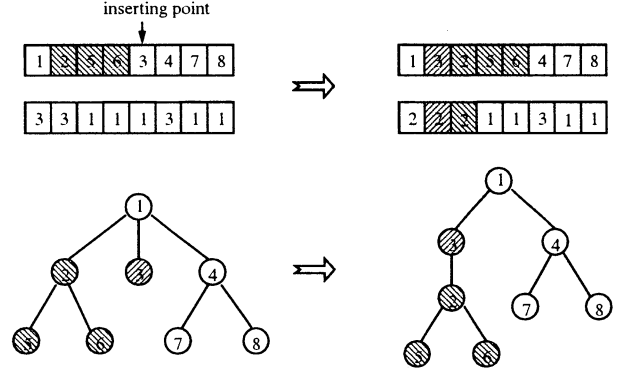


Figure 4: Illustration of insertion mutation

### 3.3 Modification

If there are such individuals whose leaves number is not equal to the leaf-constrained value, it is necessary to modify them before evaluation. Combined with the insertion mutation operation, we can easily to modify each individual. Here, we randomly select one-leaf subtree and insert it on another randomly selected one-leaf subtree as Figure 5 shows, until all leaves number is equal to the leaf-constrained value.

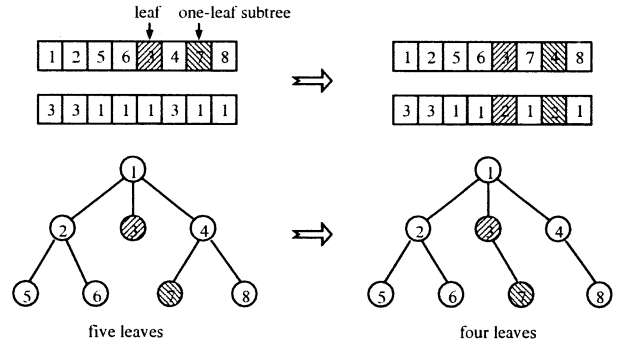


Figure 5: Illustration of leaves modification

### 3.4 Evaluation and Selection

The evaluation procedure is simply to Convert an individual into a tree according to the decoding procedure and calculate the total cost of a tree according to its objective function. As to selection, we adopt the  $(\mu + \lambda)$  - selection strategy.

## 4 Computational Experience

In this section, we compare the GA's results on the LMST problem with the lower bounds and upper bounds produced by the methods described in the previous two sections. All results are the LMST solutions

Table 1: Numerical results of the LMST problems

Problem Scale	$k$	Lower Bound	Upper Bound	GAs Result
10	3	178 (5)	192 (3)	185
20	3	262 (8)	285 (3)	274
20	5	262 (8)	280 (5)	271
30	5	365 (12)	422 (5)	401
30	8	365 (12)	416 (8)	392

Table 2: Illustration of comparison analysis

Problem Scale	$k$	(GA-LB)/LB	(UB-GA)/UB	CPU (sec.)
10	3	3.93%	3.65%	13.10
20	3	4.58%	3.86%	84.28
20	5	3.44%	3.21%	66.13
30	5	9.86%	4.98%	426.32
30	8	7.38%	5.77%	405.51

LB: lower bound, UB: upper bound.

from a class of randomly generated instances with up to 30 nodes. The cost matrix was taken as the integer uniformly and randomly distributed in the range of [10, 80]. After analyzing the parameters setting for the GA approach on the LMST problem, we found that the best settings for them are: mutation rates for exchange and inversion operations are 0.2 and 0.8, respectively, and crossover rate is 0.6.

Based on the above analysis together with the population size being 200, we figured out the following numerical examples with different leaf-constraints and tested them with the proposed GA approach.

The figures in parenthesis represent the number of leaves. Table 1 shows that the GA's results are all within the interval of their lower and upper bounds. It means that we can simply obtain a better solution of the LMST problem by the proposed GA approach.

If we consider the rate of the GA's results to their lower bound and upper bound, respectively, together with their CPU time on EWS4800/360PX workstation, Table 2 clearly illustrates the effectiveness and efficiency of the proposed GA approach. All the results are on average from 100 randomly generated LMST problems.

## 5 Conclusion

In this paper, we developed a new approach to deal with the LMST problem by using genetic algorithms. The new tree encoding has good property in keeping the heritage from parents, which provides greater chance to evolve to a highly fit structure or optimal solution in the evolutionary process. Compared with its lower and upper bounds, the numerical example shows the effectiveness of the proposed GA approach on the LMST problem.

## Acknowledgment

This research work was partially supported by the In-

ternational Scientific Research Program No. 10044173 (1998.4 – 2001.3) Grant-in-Aid for Scientific Research by the Ministry of Education, Science and Culture of the Japanese Government.

## References

- [1] Fernandes L.M. and Gouveia L.(1998), Minimal spanning trees with a constraint on the number of leaves. *European Journal of Operational Research*, **104**: 250-261.
- [2] Gen M. and Cheng R.(1997), *Genetic Algorithms and Engineering Design*. John Wiley & Sons, New York.
- [3] Zhou G. and Gen M.(1997), Approach to the degree-constrained minimum spanning tree problem using genetic algorithms. *Engineering Design and Automation*, **3/2**: 156-165.
- [4] Zhou G. and Gen M.(1997), A note on genetic algorithm approach to the degree-constrained spanning tree problems. *Networks*, **30**: 105-109.
- [5] Zhou G. and Gen M.(1998), An effective genetic algorithm approach to the quadratic minimum spanning tree problem. *Computer & Operations Research*, **25/3**: 229-237.
- [6] Zhou G. and Gen M.(1998), Genetic algorithm approach on multi-criteria minimum spanning tree problem. *European Journal of Operational Research*, in the forthcoming issue.
- [7] Zhou G. and Gen M.(1998), A new tree encoding for the degree-constrained spanning tree problems. *Soft Computing*, in the forthcoming issue.
- [8] Prim R.C.(1957), Shortest connection networks and some generalizations. *Bell Sys. Tech. J.*, **36**: 1389-1401,

# A Spanning Tree-based Genetic Algorithm for Reliable Multiplexed Network Topology Design

Jong Ryul Kim, Mitsuo Gen, and Kenichi Ida

Department of Industrial & Information Systems Engineering

Graduate School of Engineering

Ashikaga Institute of Technology

Ashikaga 326-8558, Japan

E-mail: {jrkim gen ida}@genlab.ashitech.ac.jp

## Abstract

Increasing attention is being recently devoted to various problems inherent in the topological design of distributed telecommunication networks. The topological structure of these networks can be based on a single central office and on a number of multiplexing centers. Especially, considering the high cost of the fiber optic cable, the network architecture is composed of a spanning tree. The optimization problem is reduced to choose the number and locations of the multiplexing centers, with network reliability constraints.

In this paper, a spanning tree-based GA is presented for network topology design of wide-band communication networks, considering the network reliability related to the probability of failures. We also employ the *Prüfer number* and clustering string in order to represent chromosomes. Finally, we get some experiments in order to certify the quality of the networks designs obtained by using spanning tree-based GA.

**Keywords:** Spanning Tree-based GA, Network Topology Design, Network Reliability

## 1. Introduction

Topology design problems for the communication networks system have been taken attentions by many related researchers, such as network designers, network analysts, network administrators, and so on, according to the scale of communication networks. Generally, large networks are usually comprised of a collection of networks tied together by a backbone. Also, recently, when networks are construct, fiber-optic cable is used because of its potentially limitless capabilities: huge bandwidth (nearly 50 Tps), low signal attenuation (as low as 0.2 dB/km), low signal distortion, and low power requirement, low material usage, and small space requirement<sup>6</sup>.

Increasing attention is being recently devoted to various problems inherent in the topological design of distributed telecommunication networks. The topological structure of these networks can be based on a single central office and on a number of multiplexing centers. Especially, considering the high cost of the fiber optic cable, the network architecture is composed of a spanning tree. The optimization problem is reduced to choose the number and locations of the multiplexing centers<sup>7</sup>, with network

reliability constraints.

Recently, genetic algorithms (GAs) have got a great advancement in related research fields, such as network optimization problem, combinatorial optimization, multi-objective optimization, and so on. Also, GA has received a great deal of attention about its ability as optimization techniques for many real-world problems.

In this paper, we propose a spanning tree-based GA for network topology design with fiber-optic cable, considering network reliability that means the probability of all operative nodes being connected. We also use the Prüfer number for multiplexing centers and exchange, and also employ a clustering string for users, in order to represent a candidate solution of network topology design. Finally, we get some experiments and discuss about the results which we get using spanning tree-based genetic algorithm for reliable multiplexed network topology design

## 2. Problem Formulation

The network design problem is represented by the overall cost of the network to be minimized with constraints of network reliability. It can be described as the sum of the following four components<sup>7</sup>:

- (1) the cost between users and multiplexing center

$$C_U = \sum_{i=1}^n \sum_{j=1}^m c_{1ij} \cdot x_{ij} + \sum_{i=1}^n c_{1i}^u$$

- (2) the cost of the multiplexing centers

$$C_M = \sum_{i=1}^m \sum_{j=1}^4 c_j^m \cdot \left\lceil \frac{N_{ij} + (S_j - 1)}{S_j} \right\rceil$$

- (3) the cost between multiplexing centers, including exchange

$$C_T = \sum_{i=1}^m \sum_{j=1}^m c_{2ij} \cdot y_{ij} + \sum_{i=1}^m 2 \cdot c_{k_i}^t$$

- (4) the cost of multiplexer/demultiplexer equipment located in the exchange

$$C_D = \sum_{i=1}^m \sum_{j=2}^5 N_{ij} \cdot c_i^d$$

In order to describe above cost functions, we define the following notations:

## Notation

$n$ : number of users distributed in the geographical area

$m$ : number of multiplexing centers

$c_{ij}$ : cost of the fiber-optic cable used to build the connection between users and multiplexing centers

$$x_{ij} = \begin{cases} 1, & \text{if the } i\text{-th user is connected to} \\ & j\text{-th multiplexing center;} \\ 0, & \text{otherwise} \end{cases}$$

$l_i$ : capacity level of  $i$ -th user; this takes values ranging from 1 to 5 if the available capacity demands are 2, 8, 34, 140, 565 [Mbit/s].

$c_i^u$ : cost of the equipment of the  $i$ -th user

$c_j^m$ : cost of the multiplexer with capacity level  $j$

$N_{ij}$ : number of connections at level  $j$  connected to the  $i$ -th multiplexing center

$S_j$ : maximum number that is capable of connecting to multiplexing center at level  $j$ ;  $S_j=16$  if  $j=1$  and  $S_j=4$  if  $j=2,3,4$

$c_{2ij}$ : cost of the fiber-optic cable used to build the connection between multiplexing centers

$$y_{ij} = \begin{cases} 1, & \text{if the } i\text{-th multiplexing center is} \\ & \text{connected to } j\text{-th one;} \\ 0, & \text{otherwise} \end{cases}$$

$k_i$ : capacity level of output flow of  $i$ -th center

$c_{k_i}^t$ : cost of the line terminal of level  $k_i$

$c_i^d$ : cost of multiplexing/demultiplexing equipment with capacity level equal to  $i$  in the exchange

Then we can describe the reliable multiplexed network topology design as follows:

$$\min C = C_U + C_M + C_T + C_D \quad (1)$$

$$\text{s. t. } \sum_{j=1}^m x_{ij} = 1, \quad i = 1, 2, \dots, n \quad (2)$$

$$\sum_{j=1}^m y_{ij} = 1, \quad i = 1, 2, \dots, m \quad (3)$$

$$R(\mathbf{x}, \mathbf{y}) > R_{lim} \quad (4)$$

where  $\mathbf{x}$  means a  $n \times m$  user incidence matrix representing the connection between users and multiplexing centers,  $\mathbf{y}$  is a  $m \times m$  center incidence matrix describing the connection between the multiplexing centers, and  $R_{lim}$  represents the requirement of network reliability.

Constraints (2) and (3) assure that each user is connected to a multiplexing center, and each center is connected to other multiplexing centers or to the exchange, that is, the candidate design have to be comprised of a spanning tree. Constraint (4) is about network reliability, which is described in next subsection in details.

## 2.1 Network Reliability Calculation

We consider, as reliability measure, the probability of all

operative nodes being connected. Now we want to calculate the reliability of a spanning tree network assuming that the reliability of its elements (nodes and links) are known. Considering the tree to be a rooted tree, we associate a state vector with the root of each of subtrees. The state vector associated with a root node contains all information about that node relevant to our calculation. We then define a set of recursion relations that yield the state vector of a rooted tree given the state of its subtrees. For subtrees considering of single nodes the state is obvious. Then we join the rooted subtrees into larger and larger and larger rooted subtrees using the recursion relations until the state of the entire network is obtained<sup>3</sup>.

We assume we have associated with each node  $i$  a probability of node failure  $p_i^f$  and a probability  $p_i^o = 1 - p_i^f$  of node being operative. Similarly, for the link  $(i, j)$  we have probabilities  $l_i^f$  and  $l_i^o$  of the link  $(i, j)$  failing and being operative respectively. We also define the following state vectors for each subtrees:  $b_i$  means the probability that all nodes in the subtree are failed,  $o_i$  means the probability that the set of operative nodes, including the root of the subtree, are connected, and  $r_i$  means the probability that the root of the subtree is failed and the set of operative nodes in the subtree is connected. For the tree with root node 1 and  $n$  nodes, we can calculate the reliability of tree as follows:

## Procedure: Tree-based Reliability Calculation

**Step 1:** Set  $r_i = 0$ ,  $o_i = p_i$ ,  $b_i = p_i^f$ ,  $i = 1, 2, \dots, n$ . Set  $i = n$ .

Go to step 2.

**Step 2:** If node  $j$  is the father node of node  $i$ , using the following recurrence relations, recalculate  $r_j$ ,  $o_j$ ,  $b_j$ :

$$\begin{aligned} r_j' &= r_j \cdot b_i + r_i \cdot b_j + o_i \cdot b_j \\ o_j' &= o_i \cdot o_j \cdot l_i^o + o_j \cdot b_i \\ b_j' &= b_i \cdot b_j \end{aligned}$$

Go to step 3.

**Step 3:** Set  $i = i - 1$ . If  $i = 1$ , go to step 4; otherwise go to step 2.

**Step 4:** Return  $r_1 + o_1 + b_1$ .

## 3. Genetic Algorithm

### 3.1 Representations and Initialization

Here we employ the spanning tree representation using Prüfer number in order to represent active LAN configurations. For a tree with  $n$  nodes, we can use only  $(n-2)$  digits permutation to uniquely represent a tree where each digit is an integer between 1 and  $n$  inclusive. This permutation is usually known as the Prüfer number<sup>5</sup>.

An example is given to illustrate this kind of encoding. The Prüfer number (1 1 2 2) of a spanning tree on a five node complete graph represented in Figure 1. The construction of the Prüfer number is described as follows: Locate the leaf node having the smallest label. In this case, it is node 3. Since node 1 is adjacent to node 3 in the tree, assign 1 to the first digit in the Prüfer number, and then

remove node 3 and the link (3, 1). Repeat the process on the subtree until link (2, 6) is left and the Prüfer number of this tree with three digits is finally produced.

On the contrary, for the Prüfer number  $P = (1\ 1\ 2\ 2)$ , the nodes 3, 4, 5, and 6 are eligible and  $\bar{P} = \{3, 4, 5, 6\}$ . Node 3 is the eligible node with the smallest label. Node 1 is the leftmost digit in  $P$ . Add link (3, 1) to the tree, remove node 3 from  $\bar{P}$  for further consideration, and remove the leftmost digit 1 of  $P$  leaving  $P = (1\ 2\ 2)$ . Node 4 is now the eligible element with the smallest label and the second node 1 is the leftmost digit in remaining  $P$ . Then add link (4, 1) to the tree, remove node 4 from  $\bar{P}$  for further consideration, and remove the digit 1 from  $P$  leaving  $P = (2\ 2)$ . Because node 1 is now no longer in the remaining  $P$ , it becomes eligible and is put into  $\bar{P} = \{1, 5, 6\}$ . Then, node 1 is the eligible node with the smallest label, add link (1, 2) to the tree, remove the leftmost digit 2 of  $P$ , and remove node 1 from  $\bar{P}$ . Now  $P = (2)$  and only nodes 5 and 6 are eligible. Add link (5, 2) to the tree, remove the last digit of  $P$ , and designate node 5 is not eligible and remove it from  $\bar{P}$ . Because node 2 is now no longer in the remaining  $P$  and is added to  $\bar{P}$ .  $P$  is now empty and only nodes 2 and 6 are eligible. Thus add link (2, 6) to the tree and stop. The tree is formed as shown in Figure 1.

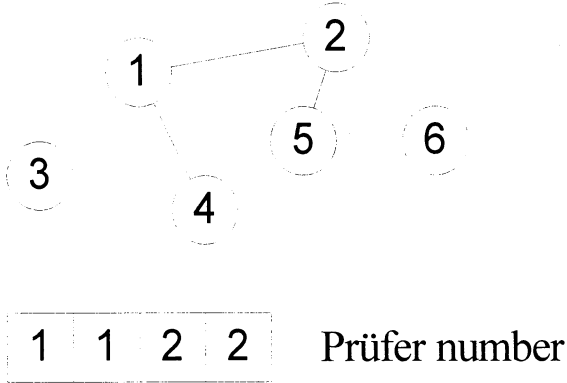


Figure 1. A tree and its Prüfer number

Prüfer number is more suitable for encoding a spanning tree, especially in some research fields, such as transportation problems, minimum spanning problems, and so on. Also, the verification for the excellence of Prüfer number is addressed by Zhou and Gen<sup>5</sup>.

Centers:	2	2	4																	
	6	7	8											...					23	
Users:	1	1	1	1	2	2	2	2	2	3	3	3	4	4	4	5	5	5	5	5

Figure 2. The Representation of Chromosome ( $m=5$ ,  $n=18$ )

We employ the following representation that has two kinds of genotype: The service centers are represented by Prüfer number and users are described by clustering string.

Therefore, we randomly generate the chromosome in the initialization process shown in Figure 2, *i.e.*, multiplexing centers are composed of  $n-2$  digits (Prüfer number) randomly generated in the range  $[1, n]$  and users are made up of  $m$  digits (clustering string) randomly generated in the range  $[1, n]$ , which mean how to allocate the users to service centers so that each user belongs to a specific service center.

### 3.2 Evaluation of Chromosome

To calculate an evaluation function  $\text{eval}(V_k)$  for reliable multiplexed network design problems, we firstly transform a chromosome  $V_k$  into user incidence matrix and center incidence matrix. Then we calculate the objective value and evaluate  $\text{eval}(V_k)$  as follows:

$$\text{eval}(V_k) = \begin{cases} 1/C, & \text{if } V_k \text{ is feasible;} \\ 1/M, & \text{otherwise} \end{cases}$$

where  $M$  is a large integer.

Form the following equation, we can keep the best chromosome  $V_k$  with largest fitness value at each generation.

$$V^* = \arg \max \{ \text{eval}(V_k) \mid k = 1, 2, \dots, \text{pop\_size} \}$$

where argmax means argument maximum.

### 3.3 Crossover and Mutation

We employed the multi-point crossover (or called uniform crossover). This type of crossover is accomplished by selecting two parent solutions and randomly taking a component from one parent to form the corresponding component of the offspring. We used here swap mutation which simply select two positions at random and swap their contents.

### 3.4 Selection

The selection used here is the method combined with the roulette wheel and elitist approach, in order to enforce the GA proposed to freely search solution space. The roulette wheel selection which is one of the fitness-proportional method is used to randomly reproduce new generation and the elitist method is employed to preserve the best chromosome for the next generation and overcome the stochastic errors of sampling.

### 3.5 Modification of Chromosome

Because of the existence of the maximum number which is capable of connecting on each center, the chromosomes generated randomly in the initial population and the offspring produced by crossover may be illegal in the sense of violating the maximum number of connection for each center. Here, we adopt the repairing strategy to modify the connection number for center in an illegal chromosome. Let  $\bar{G}$  be the set of centers whose the maximum number of connection has not been checked and modified in a chromosome. If a center  $i$  violates the

constraint with the maximum number  $g_i$  of connection for center  $i$ , then decrease the number of the center by checking the extra center and randomly replace it with another center from  $\overline{G}$ .

#### 4. Numerical Example

The spanning tree-based GA proposed in this paper for solving the reliable multiplexed network topology design problems is implemented in C language and run on a Pentium II PC.

The performance of our approach is tested with randomly generated problem:  $n=30$ ,  $m=7$ ,  $c_{ij} \in [0, 100]$ ,  $c_{2ij} \in [100, 300]$ ,  $c_{i_i}^u \in \{57, 84, 132, 216, 337\}$ ,  $c_j^m \in \{117, 167, 195, 246\}$ ,  $c_{k_i}^l \in \{56, 74, 86, 114, 156\}$ ,  $c_i^d \in \{26, 35, 49, 58, 65\}$ . Then we set the operative probability of multiplexing centers and users as 0.95, the operative probability of users, the probability between centers, and the probability between centers and users as 0.9.

The parameters for spanning tree-based GA are set as follows: pop\_size=40, max\_gen=2000, pC=0.6, pM=0.3. We got experiments 20 times.

The result of our experiments is shown in Figure 3. Also, we can get the best solution with cost and networks reliability as follows:

Centers: 4 3 6 5 7

Users: 4 5 3 7 5 3 6 1 4 1 4 4 6 2 2 3 6 7 4 1 2 3 1 7 2 6 1 7 6 2

This solution can be represented, as shown in Figure 4.

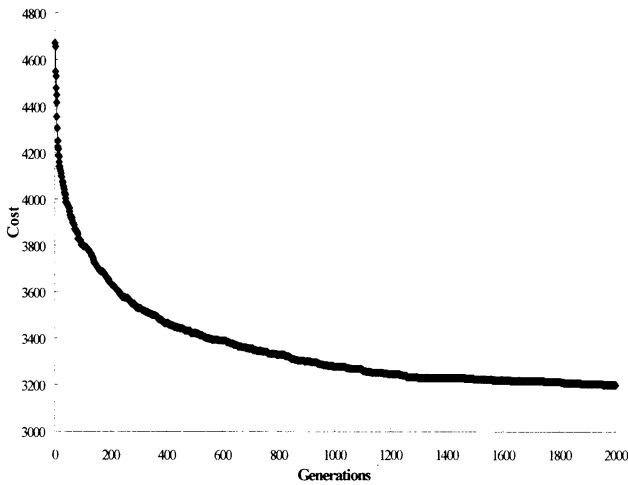


Figure 3. The Genetic Process

#### 5. Conclusion

In this paper, we proposed a spanning tree-based genetic algorithm to solve reliable multiplexed network design problems. Also, we employ tree-based network reliability calculation procedure, in order to calculate the network reliability of operative users and multiplexing centers. Because of the feature of network using fiber-optic cable in terms of cost, spanning tree configurations may be employed. From this reason, the network to be designed

has the structure of tree. Therefore, we can use recurrence relations of this network, which has tree structure.

Finally, we try on some experiments to demonstrate the effectiveness of our proposed method. From the results, we can see that our proposed GA is able to solve reliable network design problems and search optimum or near-optimum.

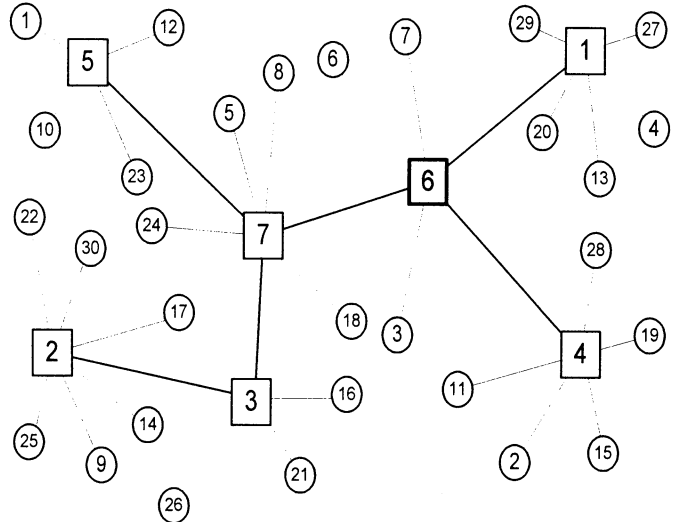


Figure 4. Network Topology with  $C=3036$  and  $R(x,y) = 0.9595$

#### Acknowledgement

This research work was partially supported by the International Scientific Research Program (No.10044173: 1998.4-2001.3) Grant-in-Aid for Scientific Research by the Ministry of Education, Science and Culture of the Japanese Government.

#### References

1. Bertsekas, D. and R. Gallager (1992). *Data Networks*, 2nd ed., Prentice-Hall, New Jersey.
2. Gen, M. and R. Cheng (1997). *Genetic Algorithms and Engineering Design*. John Wiley & Sons, New York.
3. Kershenbaum, and R. Van Slyke (1973) Recursive analysis of network reliability, *Networks*, Vol.3, pp.81-94.
4. Kershenbaum (1993). *Telecommunication Network Design Algorithms*, McGraw-Hill, Inc., New York.
5. Zhou, G., and M. Gen (1997). A Note on Genetic Algorithms for Degree-Constrained Spanning Tree Problems, *Networks*, Vol.30, pp.91-95.
6. Mukherjee B. (1997), *Optical Communication Networks*, McGraw-Hill, Inc., New York.
7. Costamagna, E., A. Fanni, and G. Giacinto (1998), A Tabu Search Algorithm for the Optimization of telecommunication networks, *European Journal of Operational Research*, Vol.106, No.2-3, pp.357-372.



## Toward Emergent Intelligence in Multiagent Learning \*

○ K. Takadama  
ATR Human  
Information Processing  
Research Labs.

T. Terano  
University of  
Tsukuba

K. Shimohara  
ATR Human  
Information Processing  
Research Labs.

K. Hori  
University of  
Tokyo

S. Nakasuka  
University of  
Tokyo

2-2 Hikaridai, Seika-cho, Souraku-gun, Kyoto 619-0288 Japan  
Tel:+81-774-95-1007 Fax:+81-774-95-1008 Email: keiki@hip.atr.co.jp

### Abstract

This paper focuses on characteristics of multiagent learning as one kinds of emergent intelligence embedded in organizations or groups, and analyzes them from the viewpoint of organizational learning. An investigation on the above characteristics has revealed the following implications: (1) there are two levels in the learning mechanisms of multiagent learning (the individual and organizational level) and each mechanism is divided into two types (single- and double-loop learning). The integration of these four learning mechanisms improves the collective performance and the ability of problem solving in multiagent organizations; (2) besides the interaction among agents, the interaction among various levels and types of learning mechanisms contributes to implementing emergent intelligence embedded in multiagent environments.

**Keywords:** emergent intelligence, organizational learning, multiagent system, learning classifier system

### 1 Introduction

What kinds of intelligence are embedded in organizations or groups? How do these kinds of intelligence emerge? They are quite important issues to be made clear. To answer these questions, not only social science but also computational science has focused on <sup>1, 8</sup>. In the field of Artificial Intelligence (AI), in particular, a lot of research on multiagents has addressed to make clear intelligence embedded in multiagent environments. Examples include multiagent reinforcement learning <sup>12</sup>, genetic algorithm (GA) <sup>5</sup>, and distributed artificial intelligence (DAI) <sup>4</sup>.

However, these kinds of research have not yet found complete answers for the above questions, because they seem to focus only on a small part of intelligence. From this fact, this paper focuses on the characteristics of multiagent learning as one of kinds of emergent intelligence embedded in multiagent environments, and starts by analyzing them from the view-

point of organizational learning (OL) <sup>2, 3, 7</sup> in organization and management science. This is because various types or levels of intelligence are embedded in OL.

This paper is organized as follows. Section 2 starts by explaining OL, and Section 3 briefly shows our model which introduces the concept of OL. Section 4 gives simulations and results, and emergent embedded intelligence is discussed in Section 5. Finally, the conclusion is given in Section 6.

### 2 Organizational Learning

Research on organizational learning (OL) has developed in the context of organization and management science, and a lot of research has focused on economical market systems or human organizations <sup>2, 3, 7</sup>. In this science, OL is roughly characterized as organizational activities for improving the organizational performance or the ability to solve problems which cannot be achieved at an individual level.

However, the features of OL somewhat differ from researcher to researcher, and consequently the definition of OL has become too general for our study. From this fact, this paper starts by selecting Kim's model <sup>6</sup> as one of the OL models to investigate emergent intelligence embedded in multiagent environments. This is because his model classifies the learning of organization in detail compared with the others <sup>1</sup>. According to Kim, OL is composed of the following four learning.

- **Individual single-loop learning:**

Through this learning, an individual improves his/her performance not by changing the contents or the amount of individual knowledge, but by utilizing them.

- **Individual double-loop learning:**

Through this learning, an individual extends the range of problem solving by changing the contents or the amount of individual knowledge, *i.e.*, creating, modifying, or removing its own individual knowledge including Macro/Meta knowledge.

<sup>1</sup>In order to analyze emergent embedded intelligence in the true sense, we must make the concept of OL operational and use them. Details will be published elsewhere <sup>11</sup>.

\* Paper submitted to the Fourth International Symposium on ARTIFICIAL LIFE AND ROBOTICS (AROB'99)

- **Organizational single-loop learning:**

Through this learning, individuals improve their performance not by changing the contents or the amount of total individual knowledge in the organization, but by utilizing them with other individuals.

- **Organizational double-loop learning:**

Through this learning, individuals extend the range of problem solving as a whole organization by changing the contents or the amount of organizational knowledge which is shared by all of the individuals, *i.e.*, creating, modifying, or removing their organizational knowledge.

The above definition mentions that (1) there are individual and organization levels in the learning and (2) each learning can be classified in terms of single type or double type.

### 3 Organizational-Learning Oriented Classifier System

Organizational-learning oriented Classifier System (OCS) <sup>9</sup> is a GBML (Genetics-Based Machine Learning) based architecture together with introducing four learning mechanisms in OL mentioned in the previous section. In OCS, each agent has these four learning mechanisms in addition to its own rules, and learns an appropriate sequence of behaviors determined by rules to solve given problems through local interactions with other agents <sup>†</sup>.

- **Reinforcement learning mechanism:**

This mechanism is performed as “individual single-loop learning” in OL, and evaluates all fired rules by changing the strength of the rules when agents solve given problems.

- **Rule generation mechanism:**

This mechanism is performed as “individual double-loop learning” in OL, and creates a new rule when all stored rules in agents do not match a current environmental state.

- **Rule exchange mechanism:**

This mechanism is performed as “organizational single-loop learning” in OL, and enables agents to exchange their rules with other agents in a particular interval.

- **Organizational knowledge reuse mechanism:**

This mechanism is performed as “organizational double-loop learning” in OL, and enables agents

to reuse the knowledge on the division of work by utilizing a set comprising each agent’s rule set as initial rules sets before other problems are solved.

## 4 Simulation

### 4.1 Problem Description

In order to investigating emergent intelligence embedded in multiagent environments, this section addresses printed circuit board (PCB) design problems as one of test-beds. The goal of the problem is to minimize the total wiring length by placing all parts appropriately. In this task domain, the parts are designed as agents in OCS, and they have some primitive behaviors such as a movement or a rotation. Using these behaviors, the parts acquire appropriate sequence of behaviors determined by their own rules to minimize the total wiring length through local interaction among parts. As an index of evaluation, we count one *step* when all parts perform one behavior, and count one *iteration* when the parts are all placed.

### 4.2 Experiment

The experiments are carried out by adding the following mechanisms one by one. Since parts cannot prepare all rules in advance, they cannot find appropriate placements only by reinforcement learning mechanism. From this factor, this paper starts the case of integrating both reinforcement learning mechanism and rule generation mechanism.

- **Reinforcement learning mechanism:** Corresponding to individual single-learning
- **Rule generation mechanism:** Corresponding to individual double-learning
- **Rule exchange mechanism:** Corresponding to organizational single-learning
- **Organizational knowledge reuse mechanism:** Corresponding to organizational double-learning

### 4.3 Results

Fig. 1 shows the change of the total wiring length of both human experts and OCS. In these figures, the horizontal axis indicates the number of the iterations and the vertical axis indicates the total wiring length. Furthermore, EXPERT indicates human experts, R indicates a Reinforcement learning mechanism, G indicates a rule Generation mechanism, X indicates a rule eXchange mechanism, and K indicates an Organizational Knowledge reuse mechanism. For example, RGXK in Fig. 1–3 indicates the case that all of four mechanisms are included.

<sup>†</sup>Detail architecture and mechanism in other papers <sup>10, 11</sup>

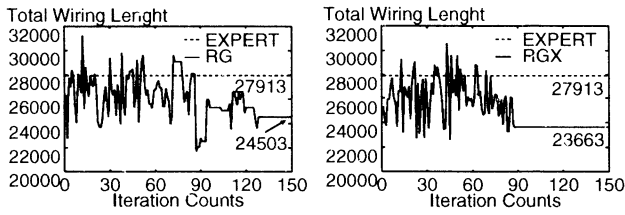


Figure 1-1

Figure 1-2

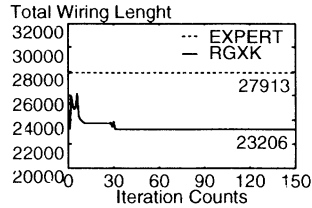


Figure 1-3

Figure 1: Iteration versus total wiring length by human experts and integration of learning mechanisms

From these figures, the total wiring length in Fig. 1-1 is converged at 24503 which is shorter than that of human experts (27913), but it needs around 130 iterations to be converged. In Fig. 1-2, both the total wiring length (23663) and the converged iterations are shorter and smaller than those in the previous figure. Furthermore, both contents in Fig. 1-3 are drastically shorter and smaller than those in Fig. 1-2.

## 5 Discussion

### 5.1 Analysis

Since we have found the effectiveness of the integration of four mechanisms described in section 4.2 through experiments, this section analyzes all of the combinations of these mechanisms for investigating the characteristics of each learning mechanism. Actually, the following 15 cases are tested with PCB design problems, and the differences in the 15 cases are compared.

- Cases 1~4 : R, G, X, K
- Cases 5~10 : RG, RX, RK, GX, GK, XK
- Cases 11~14 : RGX, RGK, RXK, GXK
- Cases 15 : RGXK

According to the results (see other papers <sup>10, 11</sup>), the following remarks are found.

- **Reinforcement learning mechanism:**

Reinforcement learning minimizes the total wiring length, but increases both the steps and the iterations. From this factor, this mechanism contributes to finding an effective solution, but it leads to a large computational complexity.

- **Rule generation mechanism:**

Parts are not able to be placed without a rule generation mechanism. This implies that this mechanism is indispensable when necessary and appropriate rules are not known beforehand.

- **Rule exchange mechanism:**

Rule exchange mechanism not only minimizes the total wiring length but also reduces the steps. From this factor, this mechanism contributes to finding effective solutions with few steps.

- **Organizational knowledge reuse mechanism:**

Although organizational knowledge reuse mechanism reduces the iterations, there is not a notable tendency in the total wiring length or the steps. However, this mechanism finds the minimum total wiring length with reducing both the steps and the iterations by integrating other mechanisms.

Furthermore, this paper examines the reasons why the above characteristics are found by analyzing the change of acquired sequences of behaviors of each part in every iteration. As a result, the analysis (see other papers <sup>11</sup>) suggests the following implications.

- Four learning mechanisms in OCS work respectively as (a) a search function, (b) a generator of search methods, (c) an entity to change the search range, and (d) an entity to effectively limit large search ranges.
- The integration of these four mechanisms contributes to finding the minimum total wiring length with reducing both the steps and iterations. This is because each mechanism works in different dimensions and makes up for the defects of an other single mechanism. In particular, the effective knowledge is utilized by *exploiting* the characteristics of a search space through the rule exchange mechanism and the organizational knowledge reuse mechanism, and the ineffective knowledge is modified/removed by *exploring* another search space through the reinforcement learning mechanism and the rule generation mechanism. This feature also indicates that the solution or the computational complexity ( $= \text{steps} \times \text{iterations}$ ) becomes worse when one of the mechanisms is missing.

### 5.2 Emergent intelligence embedded in multiagent environments

From the viewpoint of OL, the following features are found in conventional multiagent research.

- **Multiagent reinforcement learning**

Since this research focuses on how to improve a collective performance by utilizing individual knowledge prepared in advance, only individual single-loop learning is focused on in this research.

- **Genetic algorithm**

Since this research selects elite individuals with crossover and mutation operations, only individual double-loop learning and organizational single-loop learning are focused on in this research.

- **Distributed artificial intelligence**

Since this research focuses on social coordination implemented by exchanging/referring to knowledge among agents or by understanding other agents, only organizational single-loop learning is focused on in this research.

Considering these features of the conventional research from the analysis mentioned in the previous section, we have found that the above research has some limitations. This is because the above research is missing the integration with some mechanisms in OL, even though the effectiveness of the integration of four mechanisms is already shown in the previous section. From this consideration, the following conclusion is summarized.

- First, there are two levels in learning mechanisms in terms of the level between individual and organizational loop learning, and each mechanism is divided into two types in terms of the level between single- and double-loop learning. Since the solution or the computational complexity becomes worse when one of the mechanisms is missing, the integration of the four learning mechanisms in OL is required for improving the collective performance and the ability of problem solving in multiagent organizations.
- Second, an improvement of methods in each conventional research area is important for progressing the area in terms of a practical and engineering use, but the iterations among learning mechanisms in addition to the iterations among agents are also important for implementing emergent intelligence embedded in multiagent environments. This implies that the integration of various levels and types of learning mechanisms besides the four learning mechanisms in OL contributes to implementing emergent embedded intelligence.

What is important to be mentioned here is that these results acquired by an analysis from the viewpoint of OL can be applied to a lot of systems of multiagent architectures.

## 6 Conclusion

This paper analyzes characteristics of multiagent learning as one kinds of emergent intelligence embedded in multiagent environments from the viewpoint of OL. The main results are summarized as follows: (1) there are two levels in the learning mechanisms of multiagent learning (the individual and organizational level) and each mechanism is divided into two types (single- and double-loop learning). The integration of these four learning mechanisms improves the collective performance and the ability of problem solving in multiagent organizations; (2) besides the interaction among agents, the interaction among various levels and types of learning mechanisms contributes to implementing emergent intelligence embedded in multiagent environments.

Future research includes the following.

- An analysis of other levels or types of learning mechanisms with other designs of learning mechanisms or with other operationalization of OL.
- An investigation on the relationship between the organizational structure and emergent intelligence embedded in multiagent environments.

## References

1. P.E. Agre and S.J. Rosenschein (1996), *Computational Theories of Interaction and Agency* The MIT Press.
2. C. Argyris and D.A. Schon (1978), *Organizational Learning*, Addison-Wesley.
3. M.D. Cohen and L.S. Sproull (1995), *Organizational Learning*, SAGE Publications.
4. L. Gasser and A. Bond (1988) *Readings in Distributed Artificial Intelligence*, Morgan Kaufman Publishers.
5. D. E. Goldberg (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley.
6. D. Kim (1993) "The Link between individual and organizational learning", *Sloan Management Review*, Fall, pp. 37-50.
7. J.G. March (1991), "Exploration and Exploitation in Organizational Learning", *Organizational Science*, Vol. 2, No. 1, pp. 71-87.
8. M.J. Prietula, K.M. Carley and L. Gasser (1998), *Simulating Organizations* AAAI Press/The MIT Press.
9. K. Takadama et al. (1998), "Printed Circuit Board Design via Organizational-Learning Agents", *Applied Intelligence*, Kluwer Academic Publishers, Vol. 9, No. 1, pp. 25-37.
10. K. Takadama et al. (1998), "Analyzing the Roles of Problem Solving and Learning in Organizational-Learning Oriented Classifier System", *PRICAI '98*, to appear.
11. K. Takadama et al. (1998), "Making Organizational Learning Operational: Implication from Learning Classifier System", *Technical Report, TR-H-257*, to appear.
12. G. Weiss (1997), *Distributed Artificial Intelligence Meets Machine Learning - Learning in Multi-Agent Environments*, LNAI, Volume 1221, Springer-Verlag.

## Behaviors of Pedestrians with Learning Ability in Various Underground Areas

MeiHong Zheng<sup>†</sup>, Yoshiki Kasimori<sup>‡</sup>, Takeshi Kambara<sup>†‡</sup>

<sup>†</sup>Department of Information Network Science, Graduate School of Information Systems,  
The University of Electro-Communications, Chofu, Tokyo, 182-8585, Japan

<sup>‡</sup>Department of Applied Phys. and Chem., The University of Electro-Communications, Chofu, Tokyo, 182-8585, Japan

<sup>†</sup>znh@glia.pc.uec.ac.jp, <sup>‡</sup>kashi@nerve.pc.uec.ac.jp, <sup>†‡</sup>kambara@glia.pc.uec.ac.jp

### Abstract

We present a model describing motions of individual pedestrians to study collective behaviors of pedestrians in various underground areas. Each individual pedestrian has ability to decide own desired action depending on circumstances and own personality. In the present work the personality of pedestrian is expressed as patient and impatient character. The main frame of the presented model is constructed based on the social force model, and the individual ability to decide own desired action is realized by the counterpropagation neural network. The combination of neural network and dynamic model makes motion of pedestrians more natural and reliable in various situations. It has been shown from the computer simulation of passing of many pedestrians through a road that there exists an optimal ratio of patient person's number to impatient person's number for the passage time and amenity of pedestrians. As a result, the collective behaviors of pedestrians can be simulated reasonably by the present model.

**Key words:** pedestrian, social force model, neural network, personality of pedestrian, collective behaviors in passageway

### I. INTRODUCTION

To know how a crowd of pedestrians in panic behaves under various conditions is quit important for rescuer action and for designing arrangement of doorways in passageways.

During the past two decades, many kinds of dynamical models have been presented to describe the pedestrian's behaviors. These models provided valuable tools for designing and planning outlet of underground pedestrian's area such as subway or railroad station, big buildings and shopping malls. The early models were constructed based on the phenomenological fluid dynamical approach, or the specific kinetic theory. Recently, much attention has been attracted by microscopic approaches in which motions of pedestrians are described explicitly. Some individual models for pedestrian behaviors have been formulated on the basis of the idea that social field or social force guides behavioral change [1]. In these models, all pedestrians behave equivalently when they are in the same social field.

However, the real pedestrian's action is not equivalent, even if they are in the same situation. Any real pedestrian's action depends on her or his personality. Especially in panic, pedestrians' behavior will become quit different depending on their personality generated based on their past experiences. Therefore, The effect of the personality is essential to study collective behavior of

pedestrians.

In order to take account of the personality effect on the pedestrians' behavior, we present a model in which each individual has own neural network. Whose function has been determined based on his past experiences.

### II. MODEL OF BEHAVIOR OF PEDESTRIANS

The presented model is constructed based on the combination of social force model and neural network.

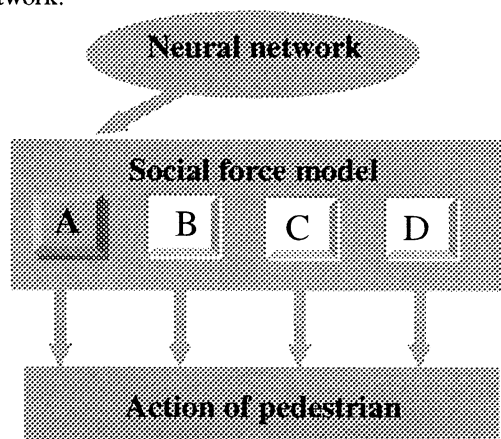


Fig.1 The structure of BP model

Figure.1 illustrates the structure of BP (behavior of pedestrians) model. In the social force model each pedestrian determines his motion taking account of 4 kinds of effects. In Figure.1 A indicates the effect of his own desire, and B and C represent the repulsive effects due to other pedestrians and walls of passage or buildings, respectively. D indicates the effect of attraction due to shopping windows or stands. The neural network determines the desired velocity of every pedestrian depending on circumstances and his or her personality.

## 1. Social force model [1]

### A. Effect of pedestrian's desire

A pedestrian  $i$  wants to reach a certain destination  $\vec{r}_i^0$ . The way will usually have the shape of a polygon with edges  $\vec{r}_i^1 \cdots \vec{r}_i^n : = \vec{r}_i^0$ . If  $\vec{r}_i^k$  is the next edge of this polygon to reach, his desired direction  $\vec{e}_i(t)$  of motion, as his present position  $\vec{r}_i(t)$  will be

$$\vec{e}_i(t) := \frac{\vec{r}_i^k - \vec{r}_i(t)}{|\vec{r}_i^k - \vec{r}_i(t)|}. \quad (1)$$

If this pedestrian's motion is not disturbed, he/she will walk in the desired direction  $\vec{e}_i(t)$  with a certain desired speed  $\vec{v}_i^0$ . A deviation of the actual velocity  $\vec{v}_i(t)$  from the desired velocity leads to a tendency to approach the desired velocity again within a certain relaxation time  $\tau_i$ . It can be described by the force

$$\vec{F}_i^0(\vec{v}_i, v_i^0 \vec{e}_i) := \frac{1}{\tau_i} (v_i^0 \vec{e}_i - \vec{v}_i). \quad (2)$$

### B. Repulsive effect due to other pedestrian

Other pedestrians influence the motion of a pedestrian  $i$ . The repulsive effect due to other pedestrian  $j$  can be represented by the force

$$\vec{F}_{ij}(\vec{r}_{ij}) := -\nabla_{\vec{r}_{ij}} V_{ij} [b(\vec{r}_{ij})]. \quad (3)$$

Where

$$2b(\vec{r}_{ij}) = \sqrt{(|\vec{r}_{ij}| + |\vec{r}_{ij} - v_j \Delta t|)^2 - (v_j \Delta t)^2}$$

$$V_{ij}(b) = V^0 \exp(-b/\sigma)$$

$$\vec{r}_{ij} := \vec{r}_i - \vec{r}_j$$

### C. Repulsive effect of the border

A pedestrian also keeps a certain distance from borders of road such as walls etc. Similarly to the repulsive effect of other pedestrian, this effect is described by the force

$$\vec{F}_{iB}(\vec{r}_{iB}) := -\nabla_{\vec{r}_{iB}} U_{iB}(|\vec{r}_{iB}|). \quad (4)$$

Here

$$U_{iB} = U^0 \exp(|\vec{r}_{iB}|/R).$$

### D. Effect of attraction

Other persons such as friends or street artists sometimes attract pedestrians. These attractive effects at place  $\vec{r}_k$  are represented by

$$\vec{F}_{ik}(|\vec{r}_{ik}|, t) := -\nabla_{\vec{r}_{ik}} w_{ik}(|\vec{r}_{ik}|, t) \quad (5)$$

Where  $\vec{r}_{ik} := \vec{r}_i - \vec{r}_k$ , and  $w_{ik}(|\vec{r}_{ik}|, t)$  is an attractive monotonic increasing potential. The attractive effect is normally decreasing with time  $t$ .

### E. Motion of pedestrian

The velocity of pedestrian  $i$  is determined by

$$\frac{d\vec{v}_i(t)}{dt} := \vec{F}_i^0(\vec{v}_i, v_i^0 \vec{e}_i) + \sum_{j \in VF} \vec{F}_{ij}(\vec{r}_{ij}) + \sum_{B \in VF} \vec{F}_{iB}(\vec{r}_{iB}) + \sum_{k \in VF} \vec{F}_{ik}(|\vec{r}_{ik}|, t) \quad (6)$$

Where  $j \in VF$  means that the summation with  $j$  runs over only pedestrian  $j$  being in the visual field of pedestrian  $i$ .

## 2. Neural network model

The magnitude and direction of desired velocity of a pedestrian  $i$  are determined by circumstance and personality of pedestrian and provided by a neural network.

In the present work, we consider two types of personalities, impatient and patient. Figure.2 illustrates two types of personality. A pedestrian is in front of the impatient person, as shown in this Figure. If her velocity is near her desired velocity, she stays in the flow line. But if her velocity is slightly smaller than her desired velocity, she looks

for a vacancy at frontal sides. If she finds a vacancy, she moves to the vacant site and walks with her desired velocity. A patient person stays usually in the flow line. But if her velocity becomes much less than her desired velocity, she passes the frontal.

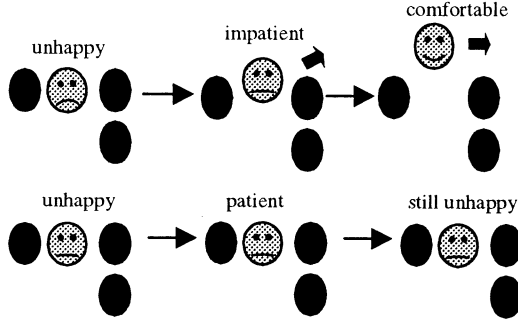


Fig.2 Two kinds of personality, patient and impatient

We represent these personalities using a counterpropagation network [2]. The network has 4 inputs, which are the personality, the difference between the desired velocity and real velocity, space on the left side and space on the right side. There are 3 outputs in this network. This network can determine 3 action patterns, which are not to evade, to evade from left and to evade from right.

In order to represent the mental condition of pedestrians in the road, we define the amenity of pedestrians' group as

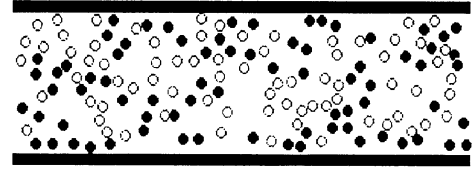
$$amt = \frac{1}{N} \sum_i \left[ \frac{1}{T_i} \sum_t \left( 1 - \frac{|v_i^0 \vec{e}_i(t) - \vec{v}_i(t)|}{v_i^0} \right) \right] \quad (7)$$

Where  $N$  is the number of the pedestrians,  $T_i$  is the time for passing from the entrance to the exit.

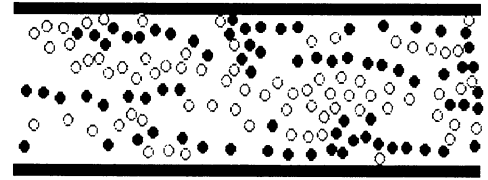
### III. SIMULATION RESULTS

In order to examine the plausibility of our model, we made some simulations of collective behavior of pedestrians in a straight road changing the width. One typical result is shown in Figure.3. Empty and solid circles represent pedestrians whose desired motion direction is right and left respectively. When the simulation starts, pedestrians can walk with a desired speed and evade each other. After several minutes, the collective behavior

becomes stable. That is, a lot of pedestrians walk along each flow line.



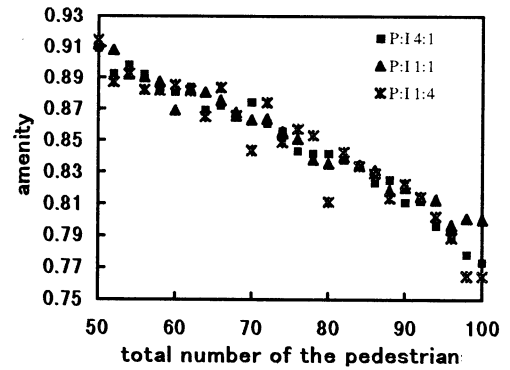
T=0: The pedestrians are randomly distributed



T=5min: the collective behavior becomes stable

Fig.3 The snapshots of pedestrian's motion obtained by BP model

In order to investigate the effect of personality on the dynamical state of pedestrian group, we calculated the amenity (Eq.7) for various



ratios of numbers of patient to impatient persons.

Fig.4 The dependence of amenity on the ratio of number of patient persons  $P$  to number of impatient persons  $I$

The Figure 4 illustrates the calculated result. The squares are the amenity in the case where the ratio of  $P$  to  $I$  is 4:1, the stars are for 1:4, and the triangles are for 1:1. In low-density cases ( $N \leq 90$ ), there is no difference among the three cases, but in high-density cases, the group with 1:1 ratio has the highest amenity. This means that there exists the

optimal ratio for the amenity.

The Figure.5 shows the amenity of the pedestrian group without neural network. In the high-density cases this amenity is lower than the amenity of the group with 1:1.

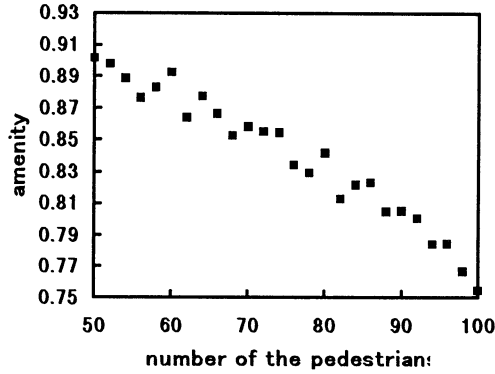


Fig.5 Amenity of the group

We calculated also the average passage time as a function of the personality ratio. Figure6 shows the result. In the low-density cases ( $N \leq 90$ ), the passage time is almost independent of the personality ratio. But in the high-density cases, the passage time becomes the shortest in the case of 1:1 ratio.

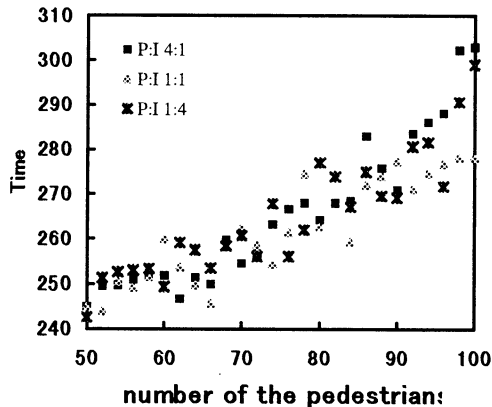
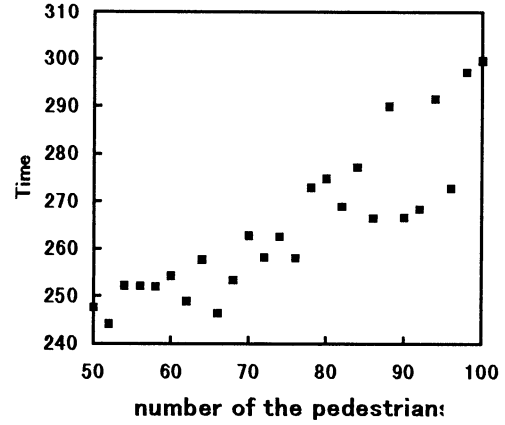


Fig.6 The dependence of the average time on the ratio of number of patient persons P to number of impatient persons I

The passage time in the social force model without neural network becomes longer in the high-density cases than the time in the case of 1:1 ratio. This result is illustrated in Figure.7.



#### IV. CONCLUSION

The collective behavior of the pedestrians could be well described by the combination of dynamic and neural network models

We have found that in the case of high-density, there exists an optimal ratio between the numbers of patient persons and impatient persons both for amenity and passage time.

We will extend this method to simulate various situations. For example, we add others personality to this model, and simulate the collective behaviors in panic.

#### REFERENCES

1. Helbing D. and Molnar P., 1995, "Social force model for pedestrian dynamics" Phys. Rev. E 51,4282-4286.
2. Hecht-Nielsen, R., 1988. Applications of counterpropagation networks, neural networks 1, 131-139.



## A Framework For Extending Classifier Systems Under Dynamic Learning Environments

Rui Jiang      Yupin Luo      Dongcheng Hu      Haifeng Xi

Dept. of Automation, Tsinghua University, Beijing 100084, P.R.China

EMail: {rjiang, luo}@mail.au.tsinghua.edu.cn

**Abstract** — A framework is proposed in this paper to extend current classifier systems to cope with dynamic environments under which learning tasks are carried out by intelligent agent(s). Within this framework, an intelligent agent leaves its own “signals” in the environment and later collects them to direct its learning process. Principles and components of the framework are outlined in the form of a *complete model*, and the *current model*, which is our present implementation of the framework, is described in detail. An experiment with the current model under a so-called Dynamic Woods1 environment is then introduced, followed by discussions over the test results. We conclude the paper by pointing out some possible improvements that can be made to the proposed framework.

**Keywords** — Classifier system, Dynamic environment, Signal store

### 1 Introduction

As a major scheme of reinforcement learning, classifier systems and their likes are explored by many researchers[Holland<sup>1</sup>, Wilson<sup>2</sup>, Dorigo et al<sup>3</sup>]. Nevertheless, in almost all of these explorations, environments are either static or single-step-reward ones, which are rare in the real world where a great number of learning environments are “dynamic” and with delayed reward. For a dynamic environment, it is insufficient to increase the exquisiteness of the agent’s internal structure, such as the supplement of a message list or temporary memory, without providing it with means by which to handle the alterations of the environment. Moreover, as can be observed in human behaviors, learners tend to facilitate the learning process by changing the environment on their own initiatives, say, leaving specific information in certain places for later reference.

With these considerations, we proposed a framework to extend classifier systems. Within this framework, an intelligent agent can leave its information in the environment and afterwards use such information to guide the learning process. The framework is implemented in Zeroth Level Classifier System(ZCS)[Wilson<sup>4</sup>] and is tested under a “Dynamic Woods1” environment. The outcome of our experiment shows that this framework can dramatically improve the learning ability of a simple intelligent agent under challenging environments.

The paper is organized as follows: In Section 2, ZCS is briefly reviewed; In Section 3, we explain the principles and components of our framework and its current implementation in ZCS; An experiment is then described in Section 4 together with an analysis of the results. In the last section, several conclusions are drawn.

### 2 Zeroth Level Classifier System

Classifier system (CS) is a rule-based learning system that was first proposed by Holland<sup>5</sup>. The system adjusts

the strengths of classifiers (condition-action rules) from environmental feedback and discovers new, plausibly better rules using genetic algorithms[Goldberg<sup>6</sup>]. Wilson suggested a Zeroth Level Classifier System(ZCS) that has been successfully applied to solve function optimization and Animat problems. ZCS is easy to understand and analyze and it has a limited learning ability which makes it an ideal testbed for our framework.

A time-step  $t$  of ZCS consists of three cycles: performance, reinforcement and discovery. In the first cycle, a message is received from the environment and matched with all the classifiers to form a match set  $M_t$ , consisting of classifiers whose condition part matches the incoming message. Then, an action  $a$  is selected from among those advocated by members of  $M_t$ , and all the members in  $M_t$  that advocated action  $a$  form the action set  $A_t$ . Finally,  $a$  is sent to the effector to be executed.

In the reinforcement cycle, the strengths of members of  $A_t$  will be adjusted according to following formula:

$$S_{A_t} \leftarrow S_{A_t} - \beta S_{A_t} + \beta r_{imm} + \beta \gamma S_{A_{t+1}} \quad (1)$$

$S_{A_t}$  denotes the total strength of members of  $A_t$ ,  $\beta$  ( $0 < \beta \leq 1$ ) and  $\gamma$  ( $0 < \gamma \leq 1$ ) are two constants,  $r_{imm}$  is the reward that the system will possibly receive from the environment because of the execution of  $a$ .

The discovery cycle is composed of a basic genetic algorithm and a covering operation. The GA is invoked with a probability to executed on the classifier population  $P$ . The *covering* operation, which creates a new classifier whose condition matches the input message and whose action is randomly generated, takes place when  $M_t$  is empty, i.e., a new input is encountered that no classifier can match it, or when the total strength in  $M_t$  is very little.

When applied in a single-step reward environment (function optimization) or a “Markovian with delayed reward” environment (Woods1), ZCS can achieve very good results and theoretically attain the optimum after sufficiently exploring the problem space.

### 3 The Framework for System Extension

The main idea behind our framework is that artificial systems cannot develop intelligent behaviors without the ability to interact with a dynamically changing environment. In the real world, we can observe a large number of learning processes during which the learners purposefully “produce” certain information to facilitate their learning, e.g., leaving special marks in the environment that will be recognized later. Therefore, in the architecture of machine learning, one can expect an improvement in system performance if the environments are taken not only as learning objects but also as learning measures.

A *complete model* based on the above ideas is presented in 3.1, which embodies the basic principles of our framework. In 3.2, the *current model* is presented, which is the implementation of the complete model.

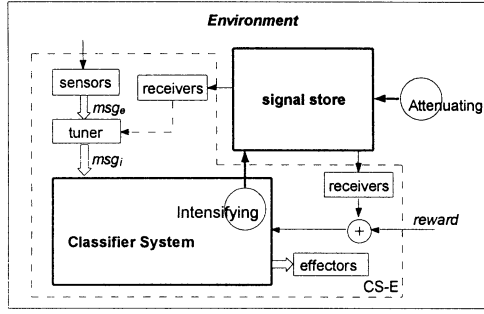


Fig. 1 The complete model

#### A. The Complete Model

The complete model is schematized in Fig. 1. The components encompassed by closed dotted line will be referred to as a Classifier System with Extension (CS-E).

The basic assumption of this model is that a classifier system can leave specific information – *signals*, in the environment. The environment will then conserve these signals using a so-called *signal store* mechanism, which can be implemented in ways dependent on the properties of the environments. Each signal possesses an initial *intensity* that will be decreased by an *attenuating* operation exercised periodically on the signal store. Signals (and their intensities) can be detected by the classifier system with special detectors called *receivers*.

During each time-step of CS-E, the environmental message ( $msg_e$ ) is detected by *sensors* and sent to a *tuner*, where an input message ( $msg_i$ ) will be formed and then forwarded to the classifier system after  $msg_e$  is “tuned” with corresponding signals. Meanwhile, an *intensifying* operation will be triggered to place a signal in the signal store or to intensify certain signals existed. After an action is taken, a reward will be adjusted by the intensity of specific signals and sent to the classifier system.

*signal* is the core conception in our framework. It “memorize” system’s behavior at certain places in the environment. Due to the tuner, previous behaviors of the CS, which are incarnated by the existence of certain signals in the environment, will be reflected in the system’s input messages, which lead to corresponding internal learning process (generation and evaluation of new classifiers). Introducing signal intensities in the rewarding scheme provides a measure to guarantee the “correct” direction of this internal learning process.

The tuner is implemented according to the system’s *sensitivity to signals*, i.e., how soon and to what extent will the signals be reflected in the input messages. A *signal attenuating speed* ( $v_{att}$ ), determined by the attenuating operation, prevents the environment from being dominated by signals. The intensifying operation determines the *signal intensifying speed*,  $v_{int}$ . The effect of signals is dependent on all these three factors.

The framework in Fig.1 can be easily extended to accommodate the situation of multiagents as illustrated in Fig. 2. The signal store, as a source of implicitly shared information, can be expected to help the formation of coordinative behaviors of multiple intelligent agent.

#### B. The Current Model

Our current implementation of the above-mentioned framework is carried out in a single ZCS under an  $l \times h$  rectilinear grid environment (Fig. 3).

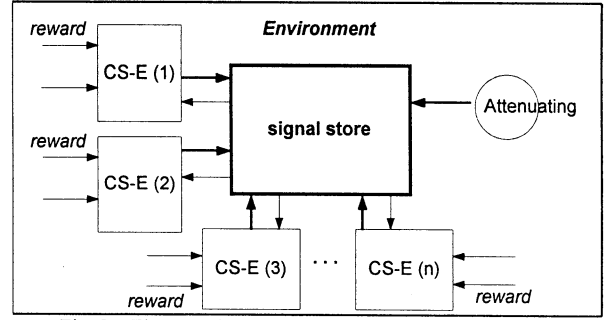


Fig. 2 The complete model with multiagents

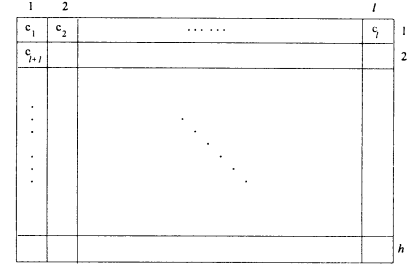


Fig. 3  $l \times h$  rectilinear grid environment

Such an environment can be defined as a set  $E = \{c_i \mid i = 1, 2, \dots, l \times h\}$ ,  $c_i$  represents a cell in the environment. Let set  $V = \{0, 1\}^r$  represent the string set consisting of all the  $r$ -bit strings, each bit of which belongs to  $\{0, 1\}$ , then we have the following definitions.

**Def. 1:** For  $\forall c_i \in E$ ,  $d_i \in V$  is defined to be  $c_i$ ’s *sensor code*;  $d_i$  can be detected by sensors and becomes a part of the message. The set  $D = \{d_i \mid i = 1, 2, \dots, l \times h\}$  represents the initial configuration of a problem to be solved.

**Def. 2:** When the classifier system is located in  $c_i$ , it can detect a message  $msg_{ei}$ , which is a concatenation of  $n$  sensor codes:  $d_{i_1}, d_{i_2}, \dots, d_{i_n} \cdot i_1, i_2, \dots, i_n \in \{1, 2, \dots, l \times h\}$ .

**Def. 3:** If  $\exists d_i \in D$ , which changes during the learning process, the environment  $E$  is said to be *dynamic*; otherwise,  $E$  is *stationary* (or *static*).

It is assumed in the current model that  $\forall c_i \in E$  can preserve one signal  $s_i$  with the intensity  $int(s_i)$ .  $int(s_i) = 0$  indicates that no signal is conserved at  $c_i$  or that  $s_i$  has “disappeared” after several times of attenuation. Assume that all the different signals comprise a *signal set*  $S$ . There is only one kind of signal, the “frequency signal”  $f$ , employed in the current model, whose intensity reflects how frequent the classifier system has visited the cell that conserves the signal. Therefore, in current model,  $S = \{f\}$ .

**Def. 4:** For  $\forall c_i \in E$ ,  $s_i \in S$  is defined to be the signal that is conserved by  $c_i$ ,  $int(s_i)$  is the intensity of  $s_i$ , which represents how frequent the classifier system has visited  $c_i$ . Initially, when time-step  $t = 0$ ,  $int(s_i) = 0$ , for  $\forall c_i \in E$ .

The following parts show how the tuner and the operation of updating and attenuating work.

**1. Tuner :** Assume that, at time-step  $t$ , the CS is located in  $c_i$ . The tuner will process those sensor codes in  $msg_e$  in the following way:

for  $k := 1$  to  $n$   
 if  $int(s_{i_k}) > C_1$ , then replace  $d_{i_k}$  with  $m$ ;  
 else leave  $d_{i_k}$  unchanged.

$m$  is the *signal code*, which is used to identify a signal. It belongs to set  $V$ , but differs from any code in  $D$ .  $C_1$  is a constant ( $C_1 \geq 0$ ) determines the sensitivity to signal.

**2. Intensifying Operation  $op_{int}$  :** When the classifier

system reaches  $c_i$  at time-step  $t$ ,  $s_i$  will be accordingly intensified as :  $int(s_i) \leftarrow u(int(s_i))$ . The form of  $u(x)$  determines  $v_{int}$ . In the current model,  $u(x)$  just takes the form of :  $u(x) = C_2x + C_3$ ,  $C_2 \geq 0$ ,  $C_3 > 0$ .

3. **Attenuating Operation**  $op_{att}$  : Every other time-step  $t_a$ , signal intensities will attenuate as follows:

for  $k:=1$  to  $l \times h$   
 if  $int(s_k) \leq C_1$ , then  $int(s_k) \leftarrow 0$ ,  
 else  $int(s_k) \leftarrow a(int(s_k))$ .

The form of  $a(x)$  and the value of  $t_a$  determine  $v_{att}$ . In the current model :  $a(x) = C_4x - C_5$ ,  $0 < C_4 \leq 1$ ,  $C_5 > 0$ .

We now extended a ZCS as follows, assuming that at the beginning of time-step  $t$ , ZCS-E is in cell  $c_k$ :

a) Discovery cycle ;  
 b) Signal operations cycle:  
 for  $\forall c_i \in E$ , execute  $op_{att}$  (every other time-steps  $t_a$ ) ;  
 for  $s_k$ , execute  $op_{int}$  ;  
 The tuner adjusts  $msg_e$  to generate input message;  
 c) Performance cycle; (Then, ZCS-E is located in  $c_j$ )  
 d) Reinforcement cycle ;  
 in this cycle, formula (1) will be revised as:

if  $S_{A_i} - \beta S_{A_i} + \beta r_{imm} + \beta \gamma S_{A_{i+1}} + \beta g(int(s_j)) \geq 0$  ,  
 then  $S_{A_i} \leftarrow S_{A_i} - \beta S_{A_i} + \beta r_{imm} + \beta \gamma S_{A_{i+1}} + \beta g(int(s_j))$ ; (2)  
 else  $S_{A_i} \leftarrow 0$ .

$g(x)$  is a function whose expression is dependent on the properties of the problem.

e)  $t \leftarrow t+1$ , goto a) .

The covering operation will generate new classifiers to match new input messages caused by the introduction of the signal code  $m$ ; the reinforcement cycle, with the effect of  $g(x)$ , tends to give the system an approximate guide on whether it is "good" or not to visit certain cells that often.

## 4 Experiment and Discussions

In this section, we will describe the experiment did and discuss about the experimental results.

### A Environment Settings

<pre> . O O F . . O O F . . O O O . . O O O . . O O O . . O O O . . . . . . . . . . . . O O F . . O O F . . O O O . . O O O . . O O O . . O O O . . . . . . . . . . . </pre>	<pre> F : food O : rock . : blank cell </pre>
--	---

Fig. 4 Layout of Dynamic Woods1

The "Dynamic Woods1(DWoods1)" environment stems directly from Woods1[Wilson<sup>4</sup>]. In a rectilinear grid, a configuration of two kinds of objects (Fig. 4) is repeated indefinitely in the horizontal and vertical directions.

The CS, or an "animat", is able to detect the sensor codes(11 for 'food', 01 for 'rock' and 00 for a blank cell) in its eight nearest cells and concatenate them by starting clockwise from north to form a 16-bit input message. The animat's available actions consist of the eight one-step moves into adjacent cells. If the adjacent cell is blank, the animat moves in; if the cell is occupied by a rock, the move is not allowed although time still elapses; if the cell contains a food, the animat moves in to "eat" the it and receives a reward. Woods1 problem is defined as: the animat is randomly placed in a blank cell, then it moves around in the

environment until a food is eaten, at which point the food immediately restores and a new problem begins. According to Def. 3, it is a static environment.

Our "DWoods1" problem is defined as: place the animat randomly in a blank cell, it starts roaming the environment until ALL the foods are eaten, then all the foods instantly re-grow and a new problem begins. In such a way, the environment will turn into a dynamic one because of the sensor code of any food-occupied cell will change after the food in it is eaten. The number of foods in DWoods1 should be finite to make the problem "solvable", therefore, the rectilinear grid in our problem is a 10×10 one containing 4 foods (Fig. 5).

### B. The Experiment

The measure of performance is an average of moving steps (the less the better) in each problem over the previous 50 problems. The following parameters and function expressions are adopted in our current model:

$m = "10"$ ;  $C_1 = 2$ ;  $u(x) = x + 1$ ;  $a(x) = x - 1$ ;  $t_a = 5$ ;  
 $g(x) = \begin{cases} -50(x - C_1) & \text{if } x > C_1; \\ 0 & \text{if } x \leq C_1. \end{cases}$

The CS-related parameters in both ZCS and ZCS-E are the same as those used in [Wilson<sup>4</sup>]. Fig.6 shows typical results. After five thousand problems, the ZCS-E's performance is about 180 steps, which is about 220 steps less than that of ZCS. The ZCS-E curve is much smoother than that of ZCS too, indicating a more stable system performance.

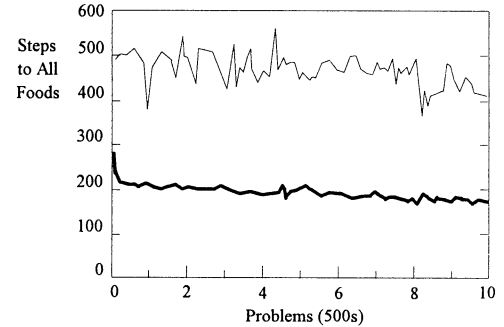


Fig. 6 Performance in Dynamic Woods1.

thin curve, ZCS; thick curve, ZCS-E. Curves are averages of 10 runs.

In the following sub-sections, we would like to discuss two phenomena observed in the experiment, and to give the reasons why ZCS-E outperformed ZCS.

### C. Looping around "Food"

In DWoods1, an obvious phenomenon can be seen in ZCS: the animat tended to visit the cell again and again even though the food in the cell had been eaten (Fig. 7).

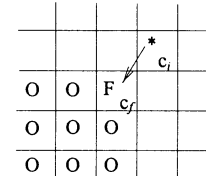


Fig. 7 Example: Looping around food

Assume that the animat(\*) is in cell  $c_i$  at time-step  $t$ , its input message is 0000000000110000 (11 means "food"). If, in  $A_i$ , there exists a classifier "0#00#0##0###0##:5 (SW)", referred to as rule1, whose action is selected, then the animat moves into  $c_j$ , eats the food and receives a reward. Because of the implicit bucket brigade mechanism implemented in the reinforcement cycle, the

animat tends to form a “habitual path” that stretches from a certain starting cell all the way to  $c_f$ . Therefore, if the animat enters a cell which is on the habitual path after it leaves  $c_f$ , it will follow the path to  $c_i$ . Because rule1 has acquired a great reward compared with the average strength of  $P$ , this “looping” behavior will have to repeat for a lot of times before the  $S_A$  at  $c_i$  is weakened enough for the covering operation to be activated to generate a new rule which leads the animat out of the habitual path. An approximate calculation shows that, if there are 5 rules in every  $A$ , rule1 will fire for an average of 13 times before the covering operation can be activated. There is yet another side-effect of this looping behavior: when the loop is finally broken, rule1, which is an effective classifier, is “destroyed” at the same time. When the animat moves away to another food’s northeast corner, rule1 is too weak to have a chance to have its action selected, the animat will probably have to learn “from the very beginning”, resulting in even more waste steps.

The poor performance described above is due to the possibility that the animat cannot sense the environmental changes correctly. Every time the animat reaches  $c_i$ , the surroundings look the same to it. In ZCS-E, however, this ambiguity is greatly decreased. With the same example, after ZCS-E loops around  $c_f$  for 2 times, the intensity of signals stored in some of the cells surrounding  $c_f$  will become so strong ( $int(s) > C_1$ ) that some corresponding sensor code in  $msg_e$  will be changed to  $m$  by the tuner; therefore, if ZCS-E once again enters  $c_i$ , either classifiers other than rule1 will be matched or the covering operation will take place. Moreover, rule1 is not weakened too much because the loop can be broken very quickly, it can still take effect elsewhere in the environment.

#### D. Looping among Blank Cells

Another phenomenon, the looping behavior among several blank cells, is observed in ZCS.

For ZCS, let’s assume that a loop forms among three adjacent cells  $c_i \rightarrow c_j \rightarrow c_k \rightarrow c_i$ , and the corresponding action sets at these cells are  $A_i$ ,  $A_j$  and  $A_k$  respectively. Since  $r_{imm}=0$  and under a reasonable assume that  $S_A$  does not change much after re-assignment, approximate equations can be written according to (1):

$$S_{A_i} \approx S_{A_i} - \beta S_{A_i} + \beta \gamma S_{A_j} \quad (3)$$

$$S_{A_j} \approx S_{A_j} - \beta S_{A_j} + \beta \gamma S_{A_k} \quad (4)$$

$$S_{A_k} \approx S_{A_k} - \beta S_{A_k} + \beta \gamma S_{A_i} \quad (5)$$

$$\text{From (4) and (5), we get } S_{A_i} \approx \gamma^2 S_{A_i} \quad (6)$$

Replace  $S_{A_j}$  in (3) with (6), we get the following approximate re-assignment:

$$S_{A_i} \leftarrow (1 - \beta + \beta \gamma^3) S_{A_i} \quad (7)$$

Similar re-assignments exist for  $S_{A_j}$  and  $S_{A_k}$ .

With the same assumption, apply the above inferring process to (2), we can get the re-assignments for ZCS-E:

$$S_{A_i} \leftarrow (1 - \beta + \beta \gamma^3) S_{A_i} + \beta (\gamma^2 + \gamma + 1) g(int(s_i)) \quad (8)$$

Similar re-assignments exist for  $S_{A_j}$  and  $S_{A_k}$ .

During the first two rounds, (8) is the same as (7) because  $int(s_i)$  equals to 0; from the third round, however, (12) will take the following form:

$$S_{A_i} \leftarrow (1 - \beta + \beta \gamma^3) S_{A_i} - 50 \beta (\gamma^2 + \gamma + 1) (int(s_i) - 2) \quad (9)$$

which accelerates the weakening of  $S_{A_i}$ . Furthermore, from the third round, the intensity of  $s_i$ ,  $s_j$  and  $s_k$  becomes so strong as to exceed  $C_1$ , which results in the changing of  $msg_{ei}$ ,  $msg_{ej}$ ,  $msg_{ek}$ , and, therefore, in possible alterations of action sets at  $c_i$ ,  $c_j$  and  $c_k$ ; this will most probably lead to an immediate loop-breaking.

No matter in ZCS-E or in ZCS, a discount factor  $\gamma$  significantly less than 1 is necessary to alleviate the animat’s looping (or “dithering”) behavior, which was observed by Wilson<sup>4</sup> and is validated in (7) and (9).

## 5 Conclusions

A framework has been proposed for extending classifier systems under dynamic learning environments. The current implementation of our framework, ZCS-E, was tested under the Dynamic Woods1 environment, where it achieved significant performance improvement compared with the original ZCS. Discussions about the experimental results have been given for a better understanding of the framework’s working mechanism.

Several problems should be further explored. First, there are three factors in the current model, signal attenuating speed  $v_{att}$ , signal intensifying speed  $v_{int}$  and system’s sensitivity to signal, which are essential to functionality of our framework. What are their respective influence on system performance, how are they mutually affected, are there any general rule-of-thumbs in deciding on their forms, these are some points of interest in the future research. Second, there is only one kind of signal in the current model, which may not be enough when the learning problem gets more difficult or a better performance is required. Therefore, other forms of signal, say direction signal, should also be tested in the framework. Third, the framework needs to be further experimented under a multiagent environment to see how well and in what way the “signal store” can act as a source of implicitly shared information to help build coordination among multiple intelligent agents.

## References

1. J. H. Holland and J. S. Reitman. Cognitive systems based on adaptive algorithms. *Pattern directed inference systems*, pages 313-329, Academic Press, New York, 1978.
2. S. W. Wilson. Classifier systems and the Animat problem. *Machine Learning*, 2(3):199-228, 1987.
3. Marco Dorigo et al. Genetics-based machine learning and behavior-based robotics: a new synthesis. *IEEE Transactions on systems, man and cybernetics*, 23(1):141-153, Jan. 1993.
4. S. W. Wilson. ZCS: a zeroth level classifier system. *Evolutionary Computation*, 2(1):1-18, 1994.
5. J.H.Holland. Escaping brittleness: the possibilities of general-purpose learning algorithms applied to parallel rule-based systems. *Machine Learning, an artificial intelligence approach*, 2. Morgan Kaufmann, Los Altos, California, 1986.
6. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, MA, Jan. 1989.
7. S. W. Wilson. Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149-175, 1995.

## Quantum Tunneling Evolution: A Model of Life as a Global Optimization Process

Masayuki HIRAFUJI and Scott HAGAN

*Computational Modeling Lab, Dept. of Information Science, National Agriculture Research Center*

Tsukuba 305-8666 Japan

### Abstract

Genetic algorithms (GAs) might be regarded as a kind of model for life, formulated as an optimization process in complex biological systems. If the size of a gene is much less than one hundred bytes, GAs can find the optimal solution and could then be relevant as a model of life. However the size of the combinatorial searching space increases exponentially with the size of the gene. GAs (that is, conventional neo-Darwinism) are not appropriate for real life.

Since the size and mass of the bases in DNA are small enough to manifest quantum effects, we propose quantum tunneling evolution (QTE), a kind of quantum computing algorithm in DNA. DNA sequences are changed by fluctuations caused by both thermal and quantum tunneling effects. The tunneling probability is not small, and bases can move in the potential of DNA. This system evolves according to the Schrödinger equation. If stress (additional ambient fluctuations) is added to the system, the width of the barrier between bases and the height of the barrier fluctuate. The tunneling probability for bases is thereby increased. The intensity of noise inversely correlates with the fitness of the phenotype. Superposition is preserved until measurement, which is effected by the transcription process. As a result, the probability of the best-fit states increases. This process is executed in parallel in individuals with superposition. QTE can accelerate the optimization process of GAs. We have developed a numerical simulator to demonstrate QTE.

Keywords: evolution, quantum, tunneling, mutation, optimization, DNA

### 1. Introduction

Many practical problems involve optimization of nonlinear systems. Artificial neural networks (ANN) such as Hopfield's model<sup>1</sup> can solve nonlinear optimization problems, but search processes are often trapped in local optima. Stochastic search methods such as simulated annealing and genetic algorithms (GAs) can potentially find global optima. However the search speed of these methods is very slow in the worst cases (for example, like Fig. 1).

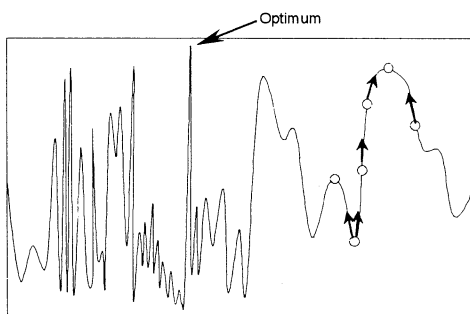


Fig. 1 An example of a difficult problem. The probability of arriving at the optimum is too small with GAs, simulated annealing, ANN and so on.

The idea of tunneling (not quantum tunneling) has been employed to avoid local minima in several algorithms, such as the dynamic tunneling algorithm<sup>2</sup> and TRUST<sup>3</sup>. We cannot get information about the global nature of evaluation functions in a classical context since optimization is restricted by locality and only local information, like the gradient of the evaluation function, is available. So global optimization is very hard in principle.

As a natural escape from classical difficulties, we

may employ quantum effects, such as are employed successfully in quantum computing<sup>4</sup>. We can immediately demonstrate new architectures of quantum neurocomputing here. For example, if we make a hardware implementation of Hopfield's model which minimizes real energy, it has the possibility of escaping from local minima by a quantum tunneling effect. This is a kind of analog quantum computer that can find global optima. However this search would take a very long time. How can we apply the quantum tunneling effect towards practical optimization? Here we propose a model of a quantum tunneling evolution (QTE). We might make use of it in an artificial setting, even if it is not employed in actual biology.

### 2. Defects of GAs as a model of life

#### 2.1 Search is too slow

Conventional GAs might be regarded as a kind of model for life, formulated as an optimization process in complex biological systems. If the gene size is not over a hundred bytes, GAs can find the optimal solution. The size of the combinatorial searching space in GAs increases exponentially with gene size. GAs (that is, conventional neo-Darwinism) cannot treat huge genomes such as those of bacteria and humans, so GAs lack some of the functions needed to solve large problems.

#### 2.2 Coding is too difficult

GAs require specific gene codes, such as Gray codes, corresponding to each problem. In actual evolution, who invents a convenient code for a developing problem? It would be much slower if GAs had to solve a problem while searching for an appropriate code.

Models based on conventional classical evolution have a speed limit. Worden defined the amount of genetic

information expressed phenotypically by GIP (Genetic Information of the Phenotype), and estimated a speed limit for evolution<sup>5</sup>. The speed limit is roughly estimated at 1/8 bit per generation, and the useful genetic information in the human brain beyond that in chimpanzee brains is at most 5 Kbytes. This seems very small, so the GIP must be highly packed. Realistic evolution algorithms must be able to find possible solutions within 350,000 generations from amongst an enormous number of possible combinations ( $10^{1200}$ ).

### 2.3 Difference of fitness is too small

To use GAs, we need to carefully prepare a fitness function. The main part of programming with GAs is to devise, using the programmer's skill and expertise, an appropriate fitness function. In actual evolution, however, fitness is evaluated by accident in a habitat and this evaluation is not sensitive to small mutations. Here no programmer attempts to improve the fitness function.

We have hypothesized that the function of the brain is to provide an accurate evaluation of fitness<sup>6</sup>. When one's situation is undesirable, the brain brings about stress. Frequently such stress facilitates one's own demise through disease. Stress must play a role not only in the termination of individuals but also in accelerating evolution.

## 3. QTM: Quantum tunneling mutation

### 3.1 Quantum tunneling in DNA

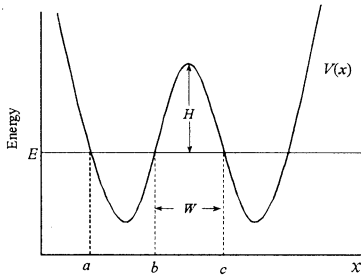


Fig. 2 Double well potential of quantum subsystem

Chemical reactions consist of both classical and quantum parts. A quantum tunneling effect allows some classically forbidden reactions to take place even near absolute zero. In the deep cold, chemical reactions occur only by tunneling, and (non-zero) reaction rate limits of many kinds of reactions are experimentally measured<sup>7</sup>. At room temperature, bases in DNA are also modified by both tunneling and classical mechanisms. The probability of tunneling per unit time,  $P_t$ , is

$$P_t = \exp\left(-\frac{1}{\hbar} \int_b^c \sqrt{2m(V-E)} dx\right) \quad (1)$$

where  $V$  is a potential;  $E$  is the energy of a base;  $m$  is the mass of a base; and  $a$ ,  $b$  and  $c$  are the locations of the classical turning points for which  $V(x)=E$  (Fig. 2). Simplifying to the case of a symmetric potential, equation

(1) reduces to

$$P_t \approx \exp\left(\frac{-\pi W \sqrt{2mH}}{4\hbar}\right) \quad (2)$$

where  $W$  is the width of the potential barrier between classical turning points and  $H$  is the height of the potential barrier. The mass,  $m$ , of bases in DNA is about 150 daltons. The intrinsic potential barrier,  $H$ , is estimated at 60 meV from measurements of the elasticity of DNA, determined directly by a mechanical method<sup>8</sup>. The width of the potential barrier,  $W$ , can be estimated at roughly 0.17 nm at most, given the pitch (0.34 nm) of the double helix structure.

The height of the potential barrier and its width fluctuate with thermal noise. The average energy of thermal noise is 25 meV, so frequently the potential barrier is lowered to 35 meV. Moreover the potential barrier fluctuates under environmental/internal stresses such as ultraviolet rays and superoxide anions, whose energies are sometimes much larger than 60 meV. In some cases the potential barrier would be diminished almost to zero and the tunneling probability will then be high.

If we assume that  $W$  is constant (0.1 or 0.17 nm), the tunneling probability is estimated as shown in Table 1. The tunneling probability is underestimated because the dynamic component due to noise (the effect of stochastic resonance) is ignored. The tunneling probability is changed dramatically by stress. Stress can be detected by this sensitive mechanism, which bears some similarity to the principal mechanism involved in STM (Scanning Tunneling Microscope). So here we obtain a first important rule, that mutation rate is regulated by stress (or fitness).

While the tunneling probability, except in the case of stress, is extremely small, the expectation probability is not small since the number of bases, the number of DNA molecules ( $10^6 - 10^{18}$ ) in an organism and the time scale ( $10^8 - 10^{14}$  sec) are enormously large. So the tunneling mutation is going on all the time.

Table 1 Tunneling Probability.

Additional noise	Potential barrier [meV]	Tunneling probability	
		$W=0.1$	$W=0.17$
none	60	3.4E-23	6.5E-39
thermal noise	35	7.0E-18	6.8E-30
stress	0.1	0.12	0.028

### 3.2 Classical mutation vs. quantum tunneling mutation

The energy of thermal noise represented by the Boltzmann distribution has a broad energy spectrum, which must be taken into account. We modeled the process as the absorption of a single thermal photon by a base. Then the tunneling probability,  $P(H/kT)$ , is

$$P(H/kT) = \frac{A}{2\zeta(3)} \int_{H/kT}^{\infty} \frac{x^2}{e^x - 1} dx$$

$$= \frac{A}{2\zeta(3)} \left\{ \frac{H^2}{k^2 T^2} \psi_1(e^{-H/kT}) + \frac{2H}{kT} \psi_2(e^{-H/kT}) + 2\psi_3(e^{-H/kT}) \right\} \quad (3)$$

where

$$\psi_0(e^z) = \frac{e^z}{1-e^z}, \quad \psi_1(e^z) = \int \psi_0(e^z) dz = -\log(1-e^z),$$

$$\psi_2(e^z) = \int \psi_1(e^z) dz, \quad \psi_3(e^z) = \int \psi_2(e^z) dz,$$

and  $\zeta(3)$  in the normalization factor is the Riemann zeta function.  $A$  is the photon absorption probability for bases, which has been estimated at 0.081<sup>9</sup>. At  $T=300$  K, the classical mutation probability (thermal transition probability) is calculated in Table 2.

The ratio of quantum mutation to classical mutation is shown in Fig. 3. Quantum mutation increases in the stressed state, and overcomes classical mutation in the region where  $H$  is smaller than about 0.1.

Table 2 Thermal Transition Probability.

Potential barrier [meV]	Transition probability
60	0.041
35	0.057
0.1	0.081

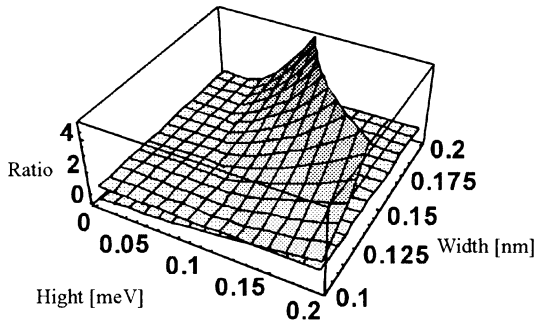


Fig. 3 Ratio of thermal transition probability to tunneling probability. The flat sheet indicates the constant plane, Ratio = 1.

### 3.3 Tunneling time for global optimization

The tunneling time,  $t_c$ , to traverse a barrier is

$$t_c = \frac{\pi}{2} W \sqrt{\frac{m}{2H}} \quad (4)$$

The tunneling time indicates the mean expectation time for a peak of the wavefunction of a base to finish traversing the barrier. Actually the shape of the wavefunction is very complicated. So it is possible to observe the base on the opposite side of the barrier within the tunneling time. Heavier particles and lower/wider potential barriers require longer tunneling times. The tunneling time of a base is given in Table 3 assuming the same variables as in 3.1. A combination of bases requires longer tunneling time.

To acquire a new epoch-making trait, multiple

bases must change simultaneously. The probability of simultaneous tunneling over  $n$  bases,  $P_s$ , is  $P_t^n$ , and the mass is increased by a factor of  $n$ . To maintain the probability in the same numerical range as that for one base (Eq. 2) we must require that the potential barrier be reduced to  $H/n^3$ . Then the tunneling time becomes  $n^2$  times by Eq. 4. This means that QTM has the potential to find the optimal solution in  $\mathcal{O}(n^2)$  time. For example, the tunneling time for a 5 kb gene is about ten seconds. And the tunneling time of the whole human genome (6 Gb) is roughly  $5 \times 10^5$  years. This result is consistent with the remarkably fast rates of evolution observed in real life. Recently another type of mutation, that is directed (adaptive) mutation, has been reported<sup>10</sup>. Directed mutation shows a mysterious ability to determine global optima and occurs without reproduction. Directed mutation may be caused by QTM.

Table 3 Tunneling Time

		Tunneling time	
		$W=0.1$	$W=0.17$
none	60	1.8E-8	3.1E-8
thermal noise	35	2.4E-8	7.5E-8
stress	0.1	4.4E-7	4.0E-7

## 4. QTE: Quantum Tunneling Evolution

### 4.1 Quantum and macroscopic complex systems

The postulated model of life is a two-layered hierarchical system that consists of a macroscopic subsystem (M-system) and a quantum subsystem (Q-system) as shown in Fig. 4. Interaction between the two subsystems is asymmetric: the Q-system affects the M-system by the transcription of information, but the physical reaction from the M-system is weak. Since the transcription process is the result of short-term interactions between DNA polymerase and DNA, the M-system cannot manipulate specific sites in the Q-system, the size of which is too small. The M-system can only affect the whole of the Q-system by using fluctuations (stress), which regulate the tunneling probability in the Q-system.

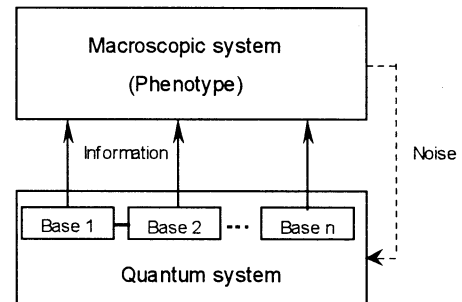


Fig. 4 A model of life as a Q-M complex system

When the M-system senses stress, the heights of

the energy barriers in the Q-system fluctuate, increasing the probability that bases in the Q-system will be transformed. Information about altered bases directly affects the M-system. So the Q-M complex system has a feedback loop which avoids higher-stress states and evolves continuously in its multi-dimensional phase space (fitness landscape). Under smaller stresses the Q-M complex system maintains its state longer, so the probability that the best-fit state is observed becomes highest.

#### 4.2 Measurement problem in DNA

Superposition in the Q-system is preserved until measurement. The wavefunction of the bases collapses by measurement. After the collapse the Q-system again starts evolving. So we get two more rules as follows.

(1) Frequently used bits are maintained and quickly revised.

The wavefunctions of bases that are frequently measured are collapsed (initialized). Simultaneously fluctuations produced by the interaction of bases and enzymes are increased, increasing the tunneling probability. On the other hand, unmeasured bases in DNA are presumably exploring the search space for new traits.

(2) Unused bits evolve through a parallel process.

This rule is hard to understand intuitively when the motion of the Q-system is slower than that of the M-system. Is the M-system in superposition if the Q-system is in superposition? This argument is related to *Schrödinger's cat*, *Stern-Gerlach measurement* and the *Elitzur-Vaidman bomb-testing problem*<sup>11</sup> show that we can get information without measurement in a quantum process. In quantum measurement experiments, we must treat non-measurement cases also as measurement cases. Extinct individuals do not exist, and this non-existence is equivalent to non-measurement of DNA. That is, cases that did not actually occur take on a role. We can therefore utilize all the combinatorial possibilities for extinct individuals to avoid a non-optimal solution. This argument may be revised as more insight into the quantum measurement problem becomes available.

#### 4.3 A sample program with QTE

If we use the above quantum functions, we can accelerate GAs. We have developed such a numerical simulator with a sample problem (traveling salesman problem) to demonstrate QTE. The fitness function is the length of a path, and the implementation is similar to that on a Hopfield model, which uses  $N^2$  quantum units (analog qbit) to represent  $N$  cities and  $N$  possible positions for each city in the sequence. Longer paths add greater noise to the quantum units. This model has 25 quantum units, which evolve in a double well potential according to the Schrödinger equation as in Fig. 5. The probability of the shortest path is highest and the shortest path was frequently observed (Fig. 6).

This model was simulated numerically (written in Java). The effect of superposition discussed in 4.1

requires an infinite number of digital processors, which was reduced into finite threads and stochastic processing.



Fig. 5 Wavefunction of a unit and the probability

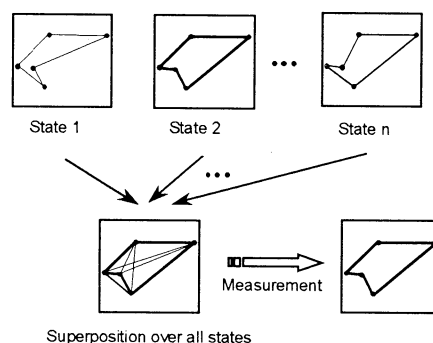


Fig.6 The shortest path is frequently observed over many states.

#### References

1. Hopfield JJ, Tank DW (1986), Computing with neural circuits: A model, *Science*, Vol. 233, pp. 625-633.
2. Yong Yao (1989), Dynamic tunneling algorithm for global optimization, *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 19, No. 5, September/October 1989, pp. 1222-1231.
3. Barhen J, Protopopescu V, Reister D (1997), TRUST: A deterministic algorithm for global optimization, *Science*, Vol. 279, pp. 1094-1097.
4. Gershenfeld NA, Chuang IL (1997), Bulk spin-resonance quantum computation, *Science*, Vol. 275, pp. 350-355.
5. Worden RP (1995), A speed limit for evolution, *J. theor. Biol.* 176, pp.137-152.
6. Hirafuji M, Hagan S (1998), An universal model for evolution of consciousness in quantum and macroscopic hierarchical systems, *Consciousness Research Abstracts*, Toward a Science of Consciousness 1998 "Tucson III", p. 144.
7. Goldanskii VI (1986), Quantum chemical reactions in the deep cold, *Scientific American*, February, pp. 38-44.
8. Smith SB, Finzi L, Bustamante C (1992), Direct mechanical measurements of the elasticity of single molecules by using magnetic beads, *Science*, Vol. 258, pp.1122-1126.
9. Vekshin NL (1997), *Energy Transfer in Macromolecules*, SPIE Optical Engineering Press.
10. Cairns J, Overbaugh J, Miller S (1988), The origin of mutants, *Nature*, Vol. 335, pp.142-145.
11. Elitzur AC, Vaidman L (1993), Quantum-mechanical interaction-free measurement, *Found. of Phys.* 23, pp. 987-997.



## Automatic Parallelization of Sequential Programs using Genetic Programming

Conor Ryan  
CSIS Dept.  
University of Limerick  
Limerick, Ireland

Laur Ivan  
CSIS Dept.  
University of Limerick  
Limerick, Ireland

### Abstract

Parallel computing has recently experienced something of a renaissance with the success of cheap alternatives to supercomputers such as Beowulf systems. Unfortunately, many of those who stand to benefit most from the speed offered by parallel architectures have large legacy systems which were designed to work on single CPU computers.

There are several powerful parallel compilers on the market (e.g. KAI C/C++, Portland Group's HPF), which are designed to recognise and exploit parallel code, however, the improvements on sequential code is usually insignificant. Obtaining a significant performance increase often leads to the rewriting from the scratch of the original code.

Our system, Paragen, is designed to alter the original program's execution flow, in order to obtain a functionally equivalent parallel code. It uses a combination of GA and GP for automatic parallelization over an infinite virtual network of processors.

## 1 Introduction

Two major strategies have been identified in the move from serial to parallel computers programming: *Rewriting* the entire application from scratch or *Parallelizing* the original sequential code.

Each of these approaches has its own advantages and disadvantages. Rewriting the entire application is very expensive from the time and financial points of view, the results are often unpredictable, and the resulted application is usually designed for a specific parallel architecture. Parallelizing is cheaper, but, using the traditional methods, there are usually restrictions to be complied with in order obtain a decent parallel code.

It is known that when parallelizing a program, the result is usually slower than rewriting (and redesigning) the application for a specific parallel architecture.

However, we believe the best trade off in terms of time spent against resulting speed can be obtained when using a GA.

## 2 Paragen

Paragen evolves sets of transformations for parallelizing the original sequential program and obtaining a functionally equivalent parallel version. It uses a *ground-up* parallelization technique: from the simplest functional blocks (**begin-end** pairs) towards the functions bodies.

### 2.1 Elements

We define the **Atom** as the smallest element recognised by Paragen. An atom is the equivalent of the generic instruction for most programming languages:

- A single loop or a sequence of consecutive loops is an atom (MetaLoop Atom),
- An instruction is an atom (Instruction Atom),
- A function call is an atom (Function Atom),
- An *if* statement is an atom (If Atom), and so on.

The atoms are processed for a specific nesting level. For example if we have:

```
0:  // level 0
1:  a=a+1;
2:  if(i=0) {
3:      // level 1
4:      for(j=1;j<10;j++) {
5:          //level 2
6:          a++;
7:      } // for (end of level 2)
8:      a-=10;
9:      b++;
10: } // if (end of level 1)
11: b++;
```

then for the level 0 the sequence processing will generate the following atoms:

```
A01: Instruction Atom ("a=a+1")
A02: If Atom ("if(i=0){...}")
A03: Instruction Atom ("b++")
```

but for level 1, the result will be:

```
A11: Loop Atom ("for(...) {...}")
A12: Instruction Atom ("a=-10")
A13: Instruction Atom ("b++")
```

Each atom takes *one timestep* to be executed.

## 2.2 Data Dependency

The main problem in parallelisation is that of data dependencies as described by Lewis<sup>3</sup>. For a generic instruction **X**, we define **X.U** as the used variables by the instruction **X**, and the modified variables by the instruction **X** as being **X.M**. Two instructions A and B are data dependent if **A.U**  $\cap$  **B.M**, **B.U**  $\cap$  **A.M** or **B.M**  $\cap$  **A.M** are not void (Lewis<sup>3</sup>).

## 3 Design

Unlike most GP implementations, Paragen does not evolve programs. Rather it is an embryonic system which progressively applies a set of evolved transformations to a serial program. The application of these transformations generate the parallel program, and, using a system we term Directed Data Dependency Analysis, these rules can subsequently be used to direct the system to the areas of the program that require analysis to prove that it is functionally identical to the original.

Paragen is designed to working in two different modes: *Atom Mode* and *Loop Mode*. The atom mode processes the input program at atom level. It has a chain of atoms as an input and it outputs the same chain with an altered execution flow. Loop mode, on the other hand, is exclusively concerned with the processing of metaloops and the extraction of loop parallelization.

### 3.1 Atom Mode Parallelization

When working in atom mode, Paragen receives the atoms from one nesting level. Then, the input is a linear chain of atoms. For this mode, Paragen evolves a binary tree of simple Atom Transformations, designed to change the original execution flow.

This mode is build based on several occam laws described by Burns<sup>1</sup>:

- $seq(A, B) = par(A, B)$  if there is no data dependencies between A and B (conversion law),
- $seq(A, B, C) = seq(A, seq(B, C))$  (associativity law),
- $seq(par(A, B), par(C, D)) = par(seq(A, C), seq(B, D))$  (distribution law).

Several classes of atom mode specific operations were generated by the OCCAM laws above:

- **Pxx** and **Sxx** class, which splits the input chain according to the percent **xx**,
- **Fxxx** and **Lxxx** class, which splits the input chain into “first and the rest” and “rest and the last” respectively, where xxx can be *PAR* or *SEQ*,
- **SHIFT** class, which delays the execution of the input chain with *one timestep*.

Each atom mode operator is an internal node in a binary tree. Any operator from those mentioned above has as input a chain of instructions and outputs two chains of instructions for their subtrees, following certain rules. The tree’s leaves are **NULL** operators and, at the end, the leaves will include the useful information about the program’s new execution flow.

The following subsection describes two operators used in Paragen, **Pxx/Sxx**, the operation of which are representative of other operators.

#### 3.1.1 Pxx/Sxx

The **Pxx/Sxx** class splits the input chain in two, according to the **xx** percentage parameter. **P** states that the instructions transmitted to the left and the right subtrees will be executed in parallel, whilst **S** means that the instructions passed to the right subtree will be executed after the chain of instructions transmitted to the left subtree have finished. For example, if we have an execution chain of 5 instructions **[ABCDE]**, then after applying a **Pxx** with **xx=40 (P40)** operator, the result will be:

Operation	Input	Output
P20	<b>[ABCDE]</b>	<b>[AB]</b>
		<b>[CDE]</b>

so **[A,B]** and **[C,D,E]** are to be executed in parallel and **[AB]** will be passed to the left subtree and **[CDE]** will be passed to the right subtree for further transformations.

Changing **P40** to an **S40** will result in the beginning of the **[CDE]** sequence after **[AB]** was finished.

### 3.2 Atom Mode Fitness Function

The fitness function has the correctness and the final speed as parameters. The correctness is the result of data dependency checking for the parallel executable atoms.

The checking is based on Directed Data Dependency Analysis, that is instead of evolving the parallel program, it is used the information from the binary tree in order to locate the areas of the program that require analysis to prove that it is functionally identical to the original.

### 3.3 Loop Mode Parallelization

Usually, the most time consuming parts of a sequential program are its iterative instructions. Loop Mode is the Paragen's section dealing with iterative instructions.

The input for loop mode is a expanded MetaLoop atom (a set of one or more consecutive loops). That is, along with the original MetaLoop atom, it receives all the atoms resulted from the next inner nesting level expansion.

We have identified several categories of loop transformations:

- single loop transformations - loop transformations for which the domain is the current loop within the metaloop,
- multiple loop transformations - loop transformations which are to be applied to several consecutive loops,
- loop order alteration transformations - loop transformations which alter the loops' execution order within the metaloop.

Below is a detailed example of a multiple loop transformation:

#### 3.3.1 Multiple Loop Transformations

The multiple loop transformations are the reason for the MetaLoop existence. A significant operator for this category is **Loop Fusion** operator. Given:

```
PARFOR(i=0;i<100;i++) {
    a[i]=a[i]*i;
    PARFOR(j=0;j<100;j++)
        b[i]=b[i]-i;
```

after applying the Loop Fusion, the result is:

```
PARFOR(i=0;i<100;i++) {
    a[i]=a[i]*i;
    b[i]=b[i]-i;
}
```

This particular operator is very useful for decreasing the communication overhead between tasks.

Another operator is the **Loop Splitting** operator, which is the opposite for loop fusion. It's usefulness is proven when having a sequential non-parallelizable loop with a parallelizable instruction within:

```
for(i=0;i<100;i++) {
    a[i]=a[i-1]*i;
    b[i]=b[i]-i;
}
```

The loop is not parallelizable because of  $a[i]=a[i-1]*i$ , but if it is split in two,

```
for(i=0;i<100;i++)
    a[i]=a[i-1]*i;
for(i=0;i<100;i++)
    b[i]=b[i]-i;
```

the second loop can be parallelized:

```
for(i=0;i<100;i++)
    a[i]=a[i-1]*i;
PARFOR(i=0;i<100;i++)
    b[i]=b[i]-i;
```

### 3.4 Loop Mode Fitness Function

While in Loop Mode, Paragen will try to apply all the operators included in the chromosome the a given MetaLoop. For now, the fitness function is relatively simple and counts the number of failures for the applied operators.

## 4 Results

To illustrate the performance of Paragen, we apply the atom mode transformations to two problems. The first contains nine data independent instructions, which gives a best case of one time step if the program were to be run in parallel. The second problem contains the same nine instructions, but these are repeated, generating a total of 18 instructions, containing nine data dependencies. This program can be executed in two time steps, however, all data dependencies must first be noted and subsequently avoided if this is to be the case.

In each of these experiments, Paragen uses a steady state population of 100, running for 10 generations.

In both cases, Paragen discovered the shortest possible program, in the first problem almost immediately. The second problem is interesting because, although there is a pattern, Paragen has no knowledge of this when generating the transformations. However, the nature of the GP engine behind Paragen is such that it discovers that the same solution to each set of instructions

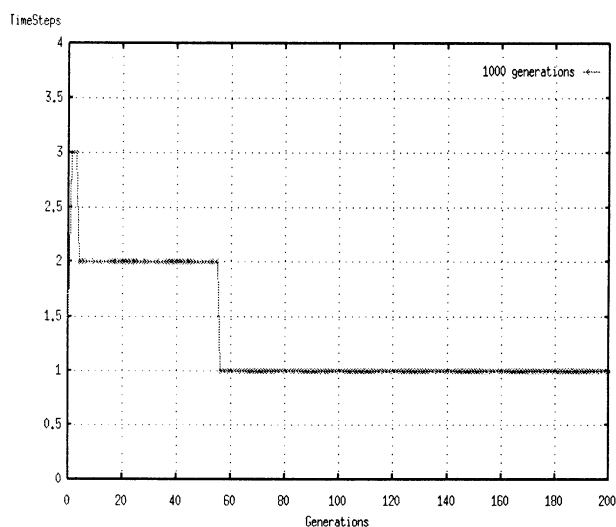


Figure 1: Paragen results with 9 data independent instructions.

## 5 Conclusions and Future Work

This paper has described an approach to automatically finding and applying suitable transformations to loops. However, due to space constraints, it is not possible to describe all possible loop transformations which have been discovered, and only a sample have been reproduced here.

Ususally, loop optimization is the key to a better performance in the conversion of sequential programs. This is why a great deal of the undergoing work is to locating more loop transformations and to integrating them in the loop mode's structure.

Another part of the research includes the implementing different breeding strategies and testing Paragen against other parallelization systems, as well as testing Paragen along with various parallel compilers.

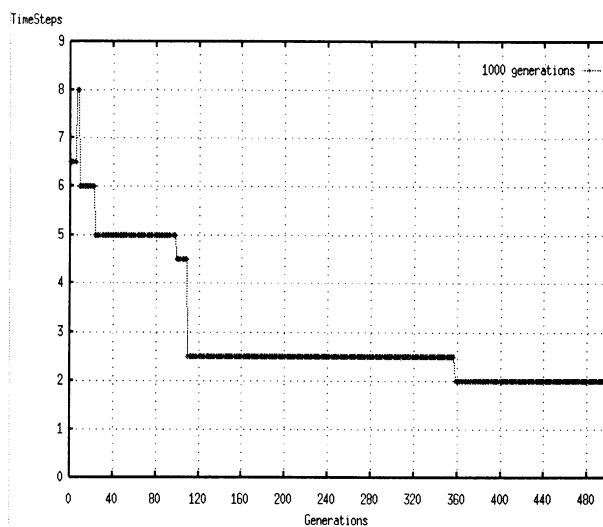


Figure 2: Paragen results with 18 instructions containing 9 data dependencies

## References

- [1] Burns, A. (1998): *Transforming Occam Programs*. In Programming Occam 2 . Addison-Wesley
- [2] Davis, L.V. (1991): *Handbook of Genetic Algorithms*. Van Nostrand Reinhold
- [3] Lewis, T. (1992) : *Introduction to Parallel Computing*. Prentice Hall
- [4] Ryan, C and Ivan, L. (1998): *Automatic Parallelization of Loops in Sequential Programs using Genetic Programming*. GP 98
- [5] Ryan, C. and Walsh, P. (1997): *The Evolution of Provable Parallel Programs*. GP 97

## A Global Search Method for All Roots of Algebraic Equations by Genetic Algorithm

S. Yamada†

I. Yoshihara†

K. Ozawa‡

K. Abe†

†Graduate School of Engineering, Tohoku University  
Sendai, 980-8579 Japan

‡Graduate School of Information Sciences, Tohoku University  
Sendai, 980-8577 Japan

{yamada, yoshiha}@largesys.ecei.tohoku.ac.jp  
ozawa@dais.is.tohoku.ac.jp, abe@abe.ecei.tohoku.ac.jp

### Abstract

Genetic algorithms are powerful for searching the optimal solution. We investigate a global search method for the roots of algebraic equations by genetic algorithm. In general, the method for solving one real root is not suitable for solving plural real roots, therefore we have proposed a new fitness function for plural real roots. In this paper, we propose a new additional strategies involving local search in order to get solutions faster.

Numerical experiments result that we can obtain all the roots for algebraic equations in consideration of the multiplicity of the roots. Finally, we confirm the effectiveness of the local search.

*Key Words* : genetic algorithm, algebraic equation, local search

## 1 Introduction

A great number of numerical methods are known to solve nonlinear equations

$$f(x) = 0, \quad x \in \mathbf{R}. \quad (1)$$

One of the most well-known methods is Newton-Raphson iteration which takes the form

$$x^{[\nu+1]} = x^{[\nu]} - \frac{f(x^{[\nu]})}{f'(x^{[\nu]})}, \quad (2)$$

$\nu = 0, 1, 2, \dots$  It is well known that if the solution which we want to get is a simple root and the initial value  $x^{[0]}$  is sufficiently close to the solution, the iteration converges quadratically, that is, the error in the current iteration is asymptotically proportional to

the square of the error in the previous one [3]. This fact shows that the iterative method has the excellent property of the local convergence. On the other hand, if we choose an unsuitable initial value which is not close enough to the solution, the approximate solutions  $y^{[0]}, y^{[1]}, \dots$  may move largely and not converge the solution which we want to get or vibrate periodically. So, it is said that there are a lot of trouble in the global convergence of Newton iteration. For the reason mentioned above, we can see that when we want to get plural roots in some interval, we have to prepare the plural initial values which are corresponding to all the solutions.

Because it is difficult to give all suitable initial values by Newton iteration, we have proposed the strategies to get all the roots in consideration of the multiplicity of the roots for algebraic equations, which are special case of nonlinear equations, by using Genetic Algorithm (GA) [4]. In this paper, we add a strategy of local search in order to get solutions faster. Finally, we perform the proposed methods and show the effectiveness.

## 2 GA for solving algebraic equations

### 2.1 A real simple root

In case of calculating one root of the equation

$$f(x) = 0 \quad (x \in \mathbf{R}) \quad (3)$$

by GA, where  $f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$ , the following method was proposed [1],[2]:

- Chromosome is defined as a real number,
- Crossover is defined as  $g_c = \lambda x_1 + (1-\lambda)x_2$ , where  $\lambda_{\min} < \lambda < \lambda_{\max}$ ,

- fitness function is defined as

$$fitness = \frac{1}{1 + |f(x)|}, \quad (4)$$

which gets the maximal value when  $x$  is the solution of the equation,

- Mutation is defined as giving a random value of the interval where all the roots exist,
- Ranking procedure is adopted as natural selection.

The proposed crossover makes us get a real number from the segment which runs between  $x_1$  and  $x_2$  on the real axis and the length of the segment depends on the values of  $\lambda_{\min}$  and  $\lambda_{\max}$ . In this paper, let  $\lambda_{\min} = -0.5$  and  $\lambda_{\max} = 1.5$ .

## 2.2 Plural real roots

In this subsection, we assume that the equation have the  $m$  real roots  $\alpha_1, \alpha_2, \dots, \alpha_m$  in consideration of the multiplicity of the roots which exist in the interval and we propose the method to get all the multiple roots.

At first, we discuss that we separate a individuals into  $m$  groups and the methods for solving a simple root apply to each group. We assume that the root  $\alpha_k$  is a simple root and two chromosomes  $x_i, x_j$  approach to  $\alpha_k$ , where  $x_i$  and  $x_j$  belong to the  $i$ -th and  $j$ -th groups respectively. In this case, each fitness of the  $i$ -th and  $j$ -th groups approaches to 1, because  $f(x_i)$  and  $f(x_j)$  approach to 0. Therefore  $\alpha_k$  is obtained as a root in the two groups. We propose a method that if we get  $x_i$  which is satisfied with  $f(x_i) = 0$  in a group, we prevent that we get  $x_j$  which is equal to  $x_i$  in other groups.

When the equation has some multiple roots, the number of the obtained solutions is less than that of the groups. This fact shows that the method is improper. In order to prevent the defect, we propose a new fitness function by the following discussion. We assume that a root  $\alpha$  is a root of the multiplicity  $n$ . At this time, we can obtain the equality

$$f(x) = (x - \alpha)^n P(x), \quad (5)$$

where  $P(\alpha) \neq 0$ . The equality (5) shows that  $f(x)$  converges to 0 with degree of  $n$ . Next, we discuss the difference product

$$g(x_i) = \prod_{j \neq i} (x_i - \hat{x}_j), \quad (6)$$

where  $\hat{x}_j$  is the representative value of the  $j$ -th group.

We assume that a chromosome  $x_i$  sufficiently approaches to  $\alpha$  which is a root of the multiplicity  $n$  and the  $l$  groups of all the groups except for the  $i$ -th group are approaching to  $\alpha$ . In this case,  $g(x_i)$  converges to 0 with degree of  $l$ , so we can get

$$F(x_i) = \frac{|f(x_i)|}{|g(x_i)|} \rightarrow \begin{cases} 0 & l \leq n-1 \\ C & l = n \\ \infty & l \geq n+1 \end{cases}, \quad (7)$$

where  $C$  is a non-zero constant. Therefore we think that it is better to define the fitness function as

$$fitness = \frac{1}{1 + F(x_i)}. \quad (8)$$

Moreover, we choose each representative value from the chromosomes which have the best five fitness values of each group at random.

## 3 Local search

Here, in order to get the roots faster, we propose a local search strategy which can provide the corrected  $x_i$  which give the minimal value of  $f(x)$  in neighborhood of  $x_i$ . The strategy is defined as following:

1. give the value of  $\varepsilon$ ,
2. calculate the tangent line of the function  $f(x)$  at  $x_i$ ,

$$h(x) = f(x_i) + f'(x_i)(x - x_i), \quad (9)$$

3. if the signs of  $h(x_i - \varepsilon)$  and  $h(x_i + \varepsilon)$  are the same

$$x_i \leftarrow \begin{cases} x_i - \varepsilon & \text{if } |h(x_i - \varepsilon)| \leq |h(x_i + \varepsilon)| \\ x_i + \varepsilon & \text{otherwise} \end{cases},$$

4. if the signs of  $h(x_i - \varepsilon)$  and  $h(x_i + \varepsilon)$  are different

$$x_i \leftarrow x_i - \frac{f(x_i)}{f'(x_i)},$$

which coincides with Newton iteration.

We can see that the strategy can give the corrected value which gives the minimal absolute value of the tangent line of the function  $y = f(x)$  at  $x_i$  inside the  $\varepsilon$ -neighborhood of  $x_i$ . In this paper,  $\varepsilon$  is determined as

$$\varepsilon = \min_i \left\{ \frac{\min_{j \neq i} |x_i - \hat{x}_j|}{2} \right\}. \quad (10)$$

Since the local search is similar to Newton iteration, we guess that when each group approaches to each corresponding root sufficiently, we can obtain all the roots faster.

Here, we can propose two application for the local search. One is to apply the local search at all generations. Another is to apply after the values of each group approach to the each corresponding root to some extent.

## 4 Numerical experiments

In this section, we apply the proposed GA to solve the equations which have only real roots. The equations are classified into three patterns by the distribution of the roots. Here, we assume that the interval of the existence of the roots is known. Moreover, the conditions of the method are given as follows:

- the number of individuals of each group is 50,
- the number of crossovers of each groups is 50,
- the number of mutations of each group is 10.
- if the absolute value of the difference product  $g(x_i)$  is less than  $10^{-20}$ , then let the value of the fitness function be 0.

Moreover, we solve the equation the following three methods:

- **method 1** no local search,
- **method 2** local search at all generations,
- **method 3** local search after the 20th generation.

### 4.1 Real simple roots (1)

At first, we deal with the equation

$$(x + 10)(x + 2)(x - 1)(x - 5)(x - 7) = 0. \quad (11)$$

The equation has 5 simple roots which are in  $[-12, 10]$ . We execute the three methods 100 times each. We show the result in Fig. 1, where the axis of ordinate shows the averages of common logarithms of errors. Here, we explain that the method to estimate their errors. We assume that the roots  $\alpha_1, \alpha_2, \dots, \alpha_n$  satisfy  $\alpha_1 \leq \alpha_2 \leq \dots \leq \alpha_n$  and let the elite of the  $i$ -th group be  $\bar{x}_i$  and assume  $\bar{x}_1 \leq \bar{x}_2 \leq \dots \leq \bar{x}_n$ . At this time, we define the error as

$$\text{error} = \sum_{i=1}^n |\alpha_i - \bar{x}_i|. \quad (12)$$

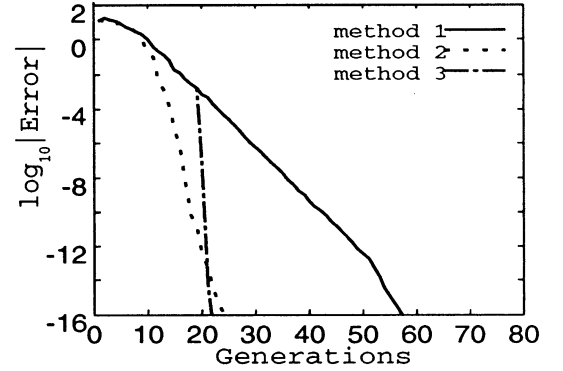


Fig. 1. Relations of generations and errors.

Fig. 1 shows that the method 2 and 3, which do the local search, are faster than the method 1, which does not do the local search.

### 4.2 Real simple roots (2)

Secondly, we solve the equation

$$x(x - 1)(x - 1.0001)(x - 1.5)(x - 2) = 0 \quad (13)$$

which has one pair of neighboring roots. The equation has 5 roots which are in  $[-2, 3]$ . We solve the equation by the three methods 100 times each. We show the result in Fig. 2.

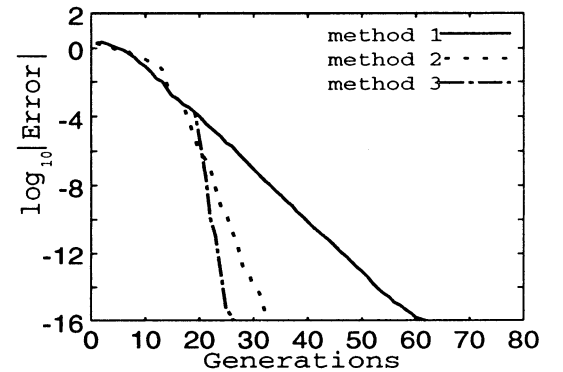


Fig. 2. Relations of generations and errors.

We can also see that the method 2 and 3 are faster than the method 1, even if the equation has a pair of neighboring roots.

### 4.3 Real multiple roots

Finally, we solve the equation

$$(x - 1)^2(x - 2)^2(x - 5) = 0 \quad (14)$$

which has one simple root and two pairs of roots of multiplicity 2 in  $[0, 6]$ . We apply the three methods to the equation 100 times each. We show the result in Fig. 3.

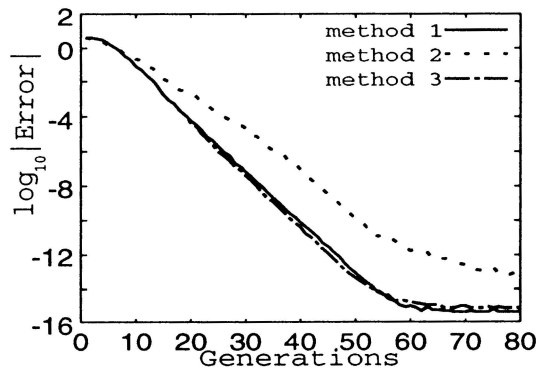


Fig. 3. Relations of generations and errors.

We can not see that the methods which have the local search strategy is always superior to the method without the local search, because the local search is almost equivalent to Newton iteration which is difficult to deal with equations which have multiple roots. However, the method 3, which is to apply the local search method after the values of each group approach to the each corresponding root to some extent, still has the excellent convergence. Moreover, we can see that the method 3 has always the superior convergence property to the others by the above experiments.

## 5 Conclusion

All the roots of algebraic equations can be hardly obtained by Newton iteration, therefore we have proposed the method to solve algebraic equations by GA. We can get all the roots of the equation using the method, even if the equation has multiple roots.

In this paper, we propose an improved method with the local search in order to get all the roots faster. The numerical experiments show that we can get all the roots faster by the proposed method than by that without local search.

## Acknowledgments

The authors wish to thank the East Japan Railway Company Endowed Chair of Large Scale System Stages Engineering in Tohoku University for supporting their research.

## References

- [1] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.
- [2] D. E. Goldberg, *Natural Frequency Calculation using Genetic Algorithms*, Proceedings of the Fifteenth Southeastern Conference on Theoretical and Applied Mechanics, pp. 94-101, 1990.
- [3] M. Mori, K. Murota and M. Sugihara, *Suchikeisan no kiso* (in Japanese), Iwanami Shoten, 1993.
- [4] S. Yamada, I. Yoshihara and K. Abe, *Global Search Methods for Solutions of Nonlinear Equations by Genetic Algorithms* (in Japanese), Proceedings of the 57th Annual Conference of IPSJ, Vol. 2, pp. 374-375, 1998.



## Optimal Control Method Using Genetic Algorithm and its Application

T. Nishimura K. Sugawara I. Yoshihara K. Abe  
Graduate School of Engineering  
Tohoku Univ.  
Sendai 980-8579, Japan  
{tune, sugawara, yoshiha}@largesys.ecei.tohoku.ac.jp,  
abe@abe.ecei.tohoku.ac.jp

### Abstract

A control method for a multi-joint manipulator whose workspace includes several obstacles is proposed. The control of manipulators requires motion planning which could be divided into two parts, 'Path planning' and 'Collision-free sequence generation'. We apply Genetic Algorithm(GA) to 'Collision-free sequence generation' to get the sequence of the movement which leads the end point (the tip of a manipulator) from the starting point to the target point without any collisions with obstacles. The effectiveness of the proposed method is evaluated through a computer simulation.

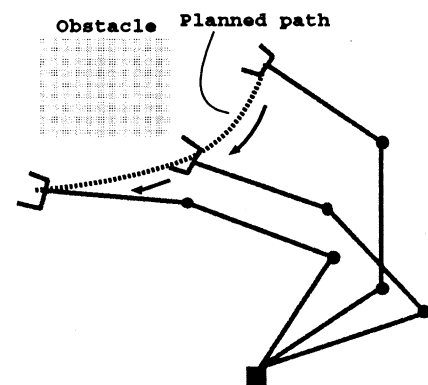


Fig.1: Motion planning for a manipulator

## 1 Introduction

Multi-joint manipulators have the advantages because they can be used in the workspace which has only a narrow space or includes a lot of obstacles in. When multi-joint manipulators are used, sometimes, the operators need to teach motion plan to the manipulators. As the number of joints increases, the difficulty of its control increases [1]. So that, teaching the motion plan is time-consuming work for operators. Even though the workspace is known previously, the motion planning for multi-joint manipulators is difficult because they can take a variety of orientations. If the collision-free motion is generated automatically, the operators do not have to teach the motion plan, and the burden of operators would be reduced.

Many methods for motion planning have been proposed. These conventional methods can be classified into two types mainly. The first type of methods is based on a method of computing an explicit representation of the manipulator configurations that would bring about a collision [2]. Some researches make this type of methods efficient using the characteristic of structure so that this type of methods cannot be

adopted or has to be modified when the structure of the manipulators changes.

The other type of methods is based on the artificial potential field concept [3]. In this kind of methods, the target point is represented by an artificial attractive potential field and the obstacles by corresponding repulsive fields so that the trajectory to the target can be generated via a flow-line tracking process with consideration to the obstacle avoidance. This type of methods has advantages of being simple and not depending on the structure of the manipulators. But, it is difficult to set up potential fields and to find an appropriate operator weight between two potential fields, for reaching the target point and for avoiding obstacles.

In this paper, we apply a method using an artificial potential field and GA with parameter tuning together to the motion planning for a manipulator. The artificial potential field gives the path from starting point to the target point while GA with parameter tuning searches collision-free sequence of the movement along the given path.

## 2 Motion planning of a multi-joint manipulator

The motion planning of a multi-joint manipulator can be realized by the combination of 'Path planning' and 'Collision-free sequence generation'. In 'Path planning', the path which leads the end point (the tip of a manipulator) from the starting point to the target point is determined. Then, in 'Collision-free sequence generation', the sequence of movement by which all links can avoid collisions with obstacles while the end point moves along the given path is searched.

### 2.1 Path planning

The artificial potential field concept is used for path planning. In general, when the artificial potential field concept is adopted, the potential field is used for both path planning and collision avoidance. Because finding an appropriate operator weight between these two potential fields and setting up potential fields itself are difficult, a potential field is used only for path planning. We assume an attractive force at the target point so the potentials are in proportion to the distance to the target. Moving to directions where the potential increase guarantees that the end point is getting close to the target point while it is avoiding local minima.

### 2.2 Collision-free sequence generation using GA

Genetic Algorithm is a search algorithm based on the mechanics of natural genetics. GA has been known its ability for global search of optimal point, and efficiently does multiple-point search[4].

Even though the path from the starting point to the target point is given, it is difficult to generate collision-free sequence of the movement along the given path, in case there are a lot of obstacles in workspace.

The position of  $i$ th joint of manipulator is given by

$$Z_{i+1} = Z_i + r_i e^{j\theta_i} \quad (1)$$

where  $r_i$  is the length of  $i$ th link,  $\theta_i$  is the angle of  $i$ th joint, and  $j$  equals to  $\sqrt{-1}$ . So, the space occupied the segment between  $\mathbf{x}_{i+1}$  and  $\mathbf{x}_i$  is represented by

$$D_i = \{Z(t) | Z(t) = tZ_{i+1} + (1-t)Z_i, 0 \leq t \leq 1\} \quad (2)$$

where  $t$  is a parameter. Then, the space where the manipulator exists is given by

$$D = D_1 \cup D_2 \cup \dots \cup D_m = \bigcup_{i=1}^m D_i \quad (3)$$

where  $m$  is the number of joints. If

		Time step				
		1	2	3	.....	n
Joint	1	13	22	-5		-17
	2	0	-11	19		14
	3	-25	17	-8		0
	.....					
	m	14	0	-3		-13

Fig.2: Structure of chromosome

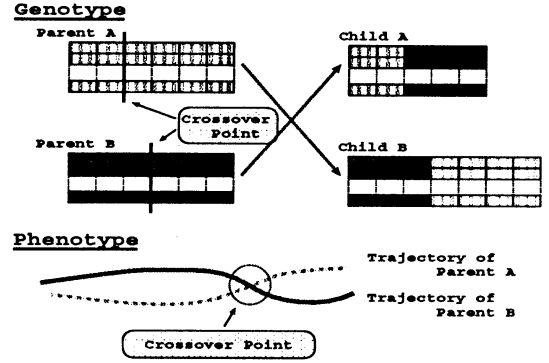


Fig.3: Crossover

$$D \subset D_f \quad (4)$$

where  $D_f$  is free-space, the space where there is no obstacle, is satisfied, the manipulator never collide with obstacles. Here,  $\theta$  is defined as  $\theta = (\theta_1, \theta_2, \dots, \theta_m)^T$  at a time-step. If  $\theta$  which satisfies the relation of (1), (2), (3), and (4) at all time-steps can be found, the collision-free sequence of movement can be realized.

We apply GA to collision-free sequence generation.

- Genetic coding

The chromosome has 2 dimensional structure (Fig.2). The length of chromosome is not fixed. Each number describes a variation in the angle of each joint at a time-step. Initial orientation of manipulator is given so that an orientation of multi-joint manipulator at a time-step can be calculated using each column of chromosome.

- Crossover

Crossover operator used here is called analogous crossover [5]. At analogous crossover operation, a pair of parents is chosen randomly and exchange their gene partly. Crossover point is decided by not the position of genotype but the function of phenotype so that the length of chromosome varies (Fig.3). The procedure of analogous crossover used in this paper is as follows.

- (1) Choose two parents (Parent A and Parent B) at random.
- (2) Choose crossover point of Parent A randomly.
- (3) Calculate the position of the end point of Parent A at the chosen crossover point.
- (4) Compare the position of the end point at each step of Parent B with calculated position of Parent A.
- (5) Select a crossover point of Parent B which is the nearest to the crossover point of Parents A.

Because parents still exist after children are born, population size becomes twice by crossover.

- Reproduction

Reproduction operator selects parents in next generation from the population which is made by crossover. One third of parents in next generation are selected from the individuals which have better fitness in order. The rest of parents are selected at random. So, population can keep the best individual without convergence.

- Mutation

Mutation operator changes the sign of a joint angle parameter. Because the sign of the parameter corresponds to the direction of a joint movement, the direction is changed by mutation. This operator does not change the absolute value of the parameter. Mutation is operated after crossover.

- Fitness

An amount of potential where there is the end point of manipulator is used as fitness. The amount of potential is determined by an artificial potential field made in 'Path planning.' This potential corresponds uniquely to the distance between the end point and the target point.

Using this fitness, the sequences of the movement which could reduce the distance to the target point is searched over generations. The fitness is related to only the distance, in other words, this fitness does not concern collision avoidance, so that generated movements tend to collide with obstacles easily. To avoid collisions, parameter tuning operation is added in GA operation as local search after crossover and mutation at each generation.

- Local search (Parameter tuning)

The parameter tuning is operated to individuals which collide with obstacles to keep search performance well. The parameters at a step which

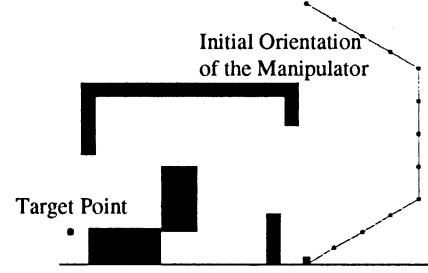


Fig.4: Workspace with obstacles

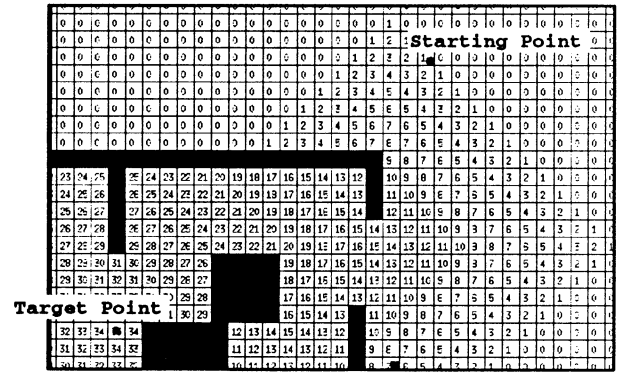


Fig.5: An artificial potential field for path planning

is just before the collision are tuned locally for a fixed period of time. Then, the parameters which can avoid the collision are adopted. If the parameters which can avoid the collision cannot be found for a fixed period of time, only the parameters before collision are adopted.

By using this strategy, we do not have to be worried about operator weights between two potential fields so that we can keep the advantages of artificial potential field concept and remove the disadvantages of this concept.

### 3 Computer simulation

To evaluate whether the motion planning for a multi-joint manipulator is realized by not using a potential field for collision avoidance, in other words, by using only a potential field which represents the distance to the target point and parameter tuning

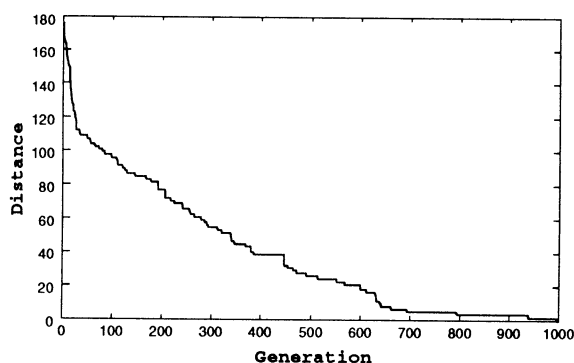


Fig.6: Experimental result

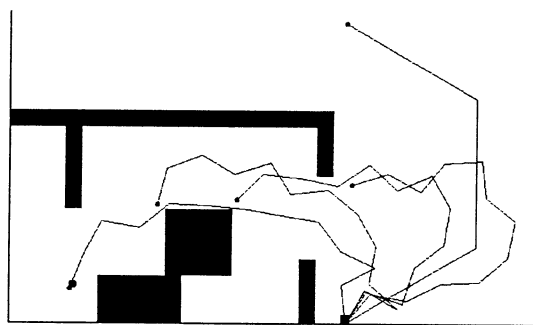


Fig.7: Trajectory of the manipulator

operation together, the proposed method is applied to motion planning for a 12-joint manipulator. The workspace has several obstacles (Fig.4), and it includes local minima. We assume that the workspace is known previously. In this workspace, the end point of manipulator has to reduce the distance to the target point with avoiding local minima, and all links has to avoid any collisions with the obstacles while the end point goes along the given path.

In 'Path planning', an artificial potential field is set up as shown in Fig.5. As it is shown in Fig.5, the artificial potential increases gradually from the starting point to the target point with avoiding local minima. Fig.6 shows the relationship between generation and distance. By using GA which includes parameter tuning operation at each generation, the best individual is getting close to the target point over generation. Fig.7 shows the trajectory of the best individual at 1000th generation of GA operation. The orientations of the manipulator at each 20 time-steps are shown. This figure shows that the distance between the end point and the target point is reduced as time-step increases while all links avoids any collisions with the obstacles.

## 4 Conclusion

In this paper, a motion planning method for a multi-joint manipulator is proposed. The proposed method uses the artificial potential field concept for 'Path planning', and the combination of GA and parameter tuning for 'Collision-free sequence generation'. GA is used to search the sequence of movement which reduces the distance to the target. The parameter tuning operation helps the sequences of movement, which are searched in GA operation, to avoid the collisions. The proposed method is applied to the motion planning of a 12-joint manipulator whose workspace has several obstacles. Then, the sequence of the movement from starting point to the target point which avoids any collisions while the end point goes along the given path is obtained.

When the fitness, related only to the distance between the end point and the target point, is used, the searched sequence of movement cannot avoid the collisions so the search will be stopped. Although this proposed method uses simple fitness, related only to the distance, parameter tuning operation keeps the performance of search well, and it is sure that the motion plan which leads the end point to the target point with avoiding collisions can be found over generations. Also, this strategy can be used even though the structure of manipulator changes.

## Acknowledgements

The authors wish to thank the East Japan Railway Company Endowed Chair of Large Scale System Stages Engineering in Tohoku University for supporting their research.

## References

- [1] John J.Craig, *Introduction to robotics : mechanics & control*, Addison-Wesley Pub. Co., 1986.
- [2] T. Lozano-Perez, "Automatic Planning of Manipulator Transfer Movements," *IEEE Tras. SMC*, Vol.11, No.10, pp.681-698, 1981.
- [3] O.Khatib, "Real-Time Obstacle Avoidance for manipulators and Mobile Robots," *The Int. Journal of Robotics Research*, Vol.5, No.1, pp.90-98,1986.
- [4] M.Hagiwara, *Neuro · Fuzzy · Genetic Algorithm* (in Japanese), Sangyo Tosho, 1994.
- [5] Edited by L.Davis, *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, 1990.

## Optimization of Foraging Behavior by Interacting Multi-robots

K. Sugawara      I. Yoshihara      K. Abe  
Graduate School of Engineering  
Tohoku University  
Sendai 980-8579, Japan  
{sugawara,yoshiha}@largesys.ecei.tohoku.ac.jp  
abe@abe.ecei.tohoku.ac.jp

### Abstract

This paper deals with an emergence of cooperative behavior among distributed autonomous robots. The robots used in our research have the simplest means of interaction to achieve a given task. In general, most of tasks can be accomplished by a sequence of basic behaviors which have some parameters. Here genetic algorithm is applied to generate a sequence of behaviors and a kind of simulated annealing is applied to determine the parameters. Effectiveness of emerging behavior is discussed through the task to collect pucks distributed in the field by focusing on the relation between the number of robots and the group efficiency.

### 1 Introduction

Social insects such as ants and bees establish well-ordered societies even in the absence of particular individual intelligence[1,2]. This is because functions are self-organized in their societies through interactions with each other. They suggest that we may be able to synthesize adaptive systems just like their societies by coordinating interactions between elements even if the function of each element is simple.

In robotics field, many researchers have been studying the multi-robot system[3,4,5,6,7,8]. These researches gave us qualitative aspect of robots behaviors, however our interest is quantitative aspects of effectiveness of multi-robot group behavior.

In previous work, we assumed a group of robots with simple interaction and discussed the effectiveness of the group through the experiments and mathematical analysis[9,10]. In this paper, we attempt the group to acquire the optimum behavior and parameters which were obtained in our previous works.

### 2 Foraging Behavior of Multi-robots

There are various tasks for which multi-robot system is useful. We investigate the collective behavior and efficiency of simple interacting multi-robots through the task to collect pucks distributed in a field (Figure 1).

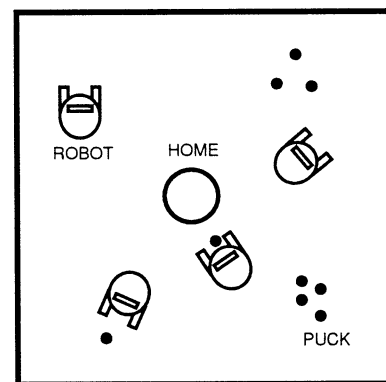


Figure 1: Foraging behavior of multi-robots.

#### 2.1 Method of interaction

Interaction process is described as follows. Each robot has a light and a pair of photo sensors. It enables robots to gather or repulse depending on the connection between the sensors and motors.

The robots interact only when a robot radiates a light signal. This duration is called as 'interaction duration' in this paper.

#### 2.2 Task as a sequence of behaviors

It was reported that a variety of complex behavior can be described by a combination of several basic

behaviors[7]. This research showed a proper behavior was selected depending on an environment and a situation. In this paper, we assume that the task can be accomplished by a sequence of basic behaviors which have some parameters. It is assumed that each robot has five basic behaviors.

- Wandering: to move straight unless meeting a puck, other robots or boundary walls.
- Broadcasting: to radiate singal light isotropically.
- Attracted: to move toward the signal source.
- Homing: to move toward home.
- Staying: to stay there.

It is noticed that each behavior has an obstacle avoidance behavior in order to avoid collision.

Foraging behavior can be described by the combination of these behaviors. The purpose of this paper is to observe the emergence of the proper sequence and parameters which maximize their efficiency (Figure 2).

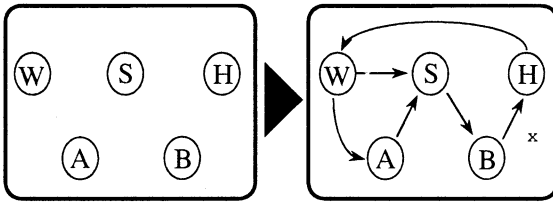


Figure 2: Schematic of the purpose of this study .

### 3 Simulation and result

#### 3.1 Method

To obtain an appropriate combination of basic behaviors, we need to generate a sequence and determine parameters included in each behavior.

- Sequence generation

We apply Genetic Algorithm(GA) to generate the behavior sequence[11]. By regarding one basic behavior as a gene, a sequence of behaviors can be considered as a chromosome. Then, optimum sequence can be obtained by following GA operation.

- Crossover

Two sequences are chosen at random as parents. A position on each sequence is determined at random, and the segments of the sequence are exchanged each other. This process brings a child(new sequence).

- Selection

Fitness of individual is determined by the number of

collected pucks during a constant period. Next generation individuals are selected by the ranking. An individuals of higher fitness survives and lower are killed.

- Mutation

One of the gene is chosen at random and changed another gene.

- Parameter determination

Each basic behavior has some variable parameters. In most case, we can consider a searching space of the parameter is not complex, because the basic behavior has just a simple function. Under this assumption, the value of parameter is determined by a kind of simulated annealing. It means that searching space which is wide at first is gradually narrowed and the optimum value is determined. In this paper, we mainly focus on the interaction duration. It is because this parameter is the most important for the effectiveness of the group.

#### 3.2 Simulation condition

We examine the efficiency of this method by computer simulation. Here a spatially discrete model is applied for rapid computation[12].

We assume a square field. Home is located at the center. The field is partitioned into  $400 \times 400$  cells. Each cell could have 3 states: 'empty', 'robot' or 'puck'. The position and the direction of each robot are calculated as continuous variables. A cell which includes a robot is occupied and has the state 'robot'. The velocity of the robots is  $10 \text{ cells/sec}$  and the angular velocity is  $120 + \Delta\theta \text{ deg/sec}$ , where  $\Delta\theta$  is a random variable uniformly distributed within  $(-5, +5)$  degrees. When they meet boundary walls and other robots, they turn to random direction. A time step of simulation is  $0.1 \text{ sec}$ . The initial location and direction of robots are uniformly random.

There is a various types of puck distribution. Here we choose a localized distribution which all pucks are located in a circle with its center at  $(180, 180)$  and with a radius of 10.

#### 3.3 Simulation result

Time evolution of gathering ability by the robots is shown in figure 3(a). A vertical axis implies the number of collected pucks during a constant period. The number of robot is 100. In initial generation, their gathering ability is low. But the ability is improved gradually and is saturated over 30 generation. This figure implies the robot system acquires a proper sequence of behaviors. Figure 3(b) shows the acquired

interaction duration in each generation. This figure shows that the interaction duration varies excessively in small number of robots. On the other hand, in case of large number of robots, the fluctuation of interaction duration is small.

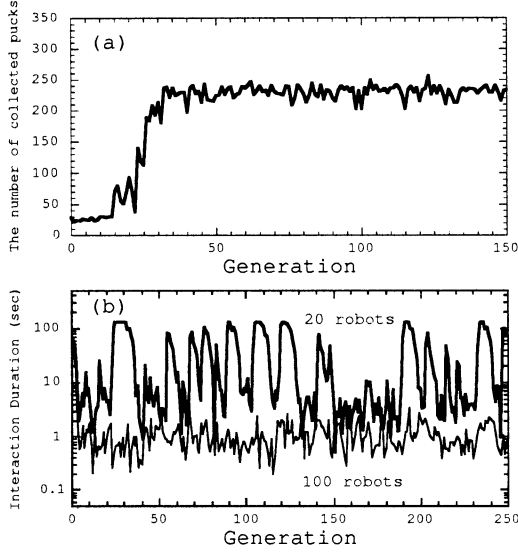


Figure 3: Time evolution of gathering ability (a) and interaction duration (b).

It was reported that the number of robots  $N$  and task completion time  $T$  has a relation  $T \sim N^\beta$  in this task[12]. A value of  $\beta$  depends on the interaction duration. Figure 4(a) shows the relation between the interaction duration and exponent  $\beta$ . There is a optimum interaction duration  $I_{opt}$ . Figure 4(b) shows the histogram of interaction durations which are shown in figure 3(b). In case of small number of robots, dispersion is large. But in case of large number of robots, dispersion is small and an average value is almost same as the obtained  $I_{opt}$ .

Figure 5 shows the relation between the number of robots and the average of acquired interaction duration. It shows that the larger the number of robot, the better the acquired interaction duration.

#### 4 Conclusion

We proposed that a task can be accomplished by a sequence of basic behaviors which include some parameters. A sequence of behaviors is considered as a chromosome by regarding each basic behavior as a

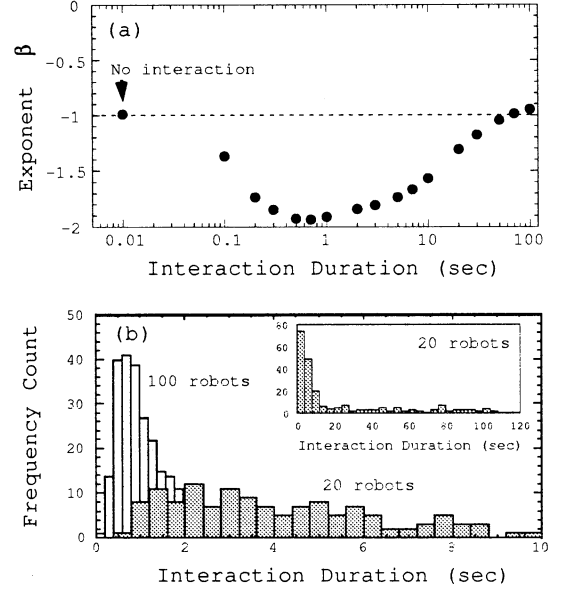


Figure 4: Acquired values of interaction duration. (a):Relation between interaction duration and exponent  $\beta$ [12]. (b):Histogram of acquired interaction duration within 250 generation.

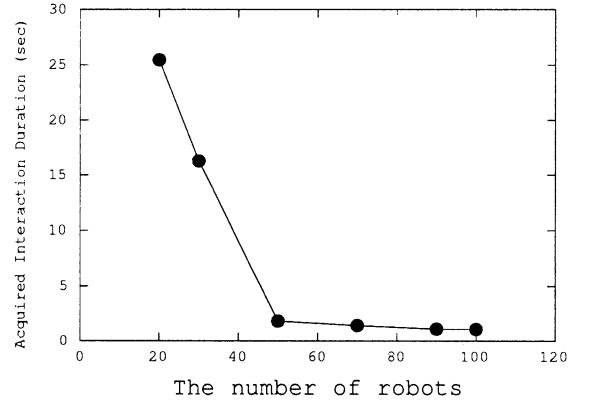


Figure 5: The relation between the number of robots and the average of acquired interaction duration.

gene. This assumption enables us to apply genetic algorithm to our robot system. Parameters included in each behavior can be optimized by a kind of simulated annealing.

We chose a foraging behavior as a task. Its characteristic is already investigated under a given condition. So we could confirm the efficiency of the proposal method referring those results.

A result of simulation shows the robots acquired a proper sequence and parameters. Focusing on the most important parameter, interaction duration, we found that the interaction duration varies excessively in small number of robots. On the other hand, the larger the number of robot is, the smaller the fluctuation of interaction duration is.

Now we are studying their behavior under more complex environment. We also examine their behavior using a real robot system.

## Acknowledgement

We would like to thank Y. Sawada and M. Sano for valuable comments and discussion. We also wish to thank the East Japan Railway Company Endowed Chair of Large Scale System Stage Engineering in Tohoku University for supporting our research.

## References

- [1] B. Hölldobler and E. O. Wilson, *JOURNEY TO THE ANTS*, Harvard University Press, (1994).
- [2] T. D. Seeley, *THE WISDOM OF THE HIVE*, Harvard University Press, (1995).
- [3] T. Sato, et al., *J. SICE*, 31, (1992)(in Japanese).
- [4] K. Asama, et al., Eds. *Distributed Autonomous Robotic System 2*, Springer-Verlag, (1996),
- [5] R. Beckers, O.E. Holland and J.L. Deneubourg, *Artificial Life IV*, MIT Press, (1994), pp. 181.
- [6] T. Balch and R.C. Arkin, *Autonomous Robots*, 1, (1994), pp. 27.
- [7] M.J. Mataric, Ph.D Thesis, MIT, (1994).
- [8] L. E. Parker, *Proc. of 1995 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, (1995) , pp. 212.
- [9] K. Sugawara and M. Sano, *Physica D*, 100, (1997), pp.343.
- [10] K. Sugawara, I. Yoshihara and K. Abe, *Proc. 3rd Int'l Symp. on Artificial life and Robotics*, (1998) pp.145.
- [11] L. Davis, *Genetic Algorithm Handbook*, Van Nostrand Reinhold, (1990).
- [12] K. Sugawara and M. Sano, *Distributed Autonomous Robotic System 2*, Springer-Verlag, (1996), pp. 233.



## Optimization of Delivery Route in a City Area using Genetic Algorithm

A.Takeda S.Yamada K.Sugawara I.Yoshihara K.Abe

Graduate School of Engineering

Tohoku University

Sendai,980-8579,JAPAN

e-mail: {takeda,yamada,sugawara,yoshiha}@largesys.ecei.tohoku.ac.jp  
abe@abe.ecei.tohoku.ac.jp

### Abstract

When we solve the delivery route optimization problem, We must regard this problem as an Asymmetric Travelling Salesman Problem (ATSP) because there are many one-way streets.

In this paper, we solve ATSP using Genetic Algorithm (GA). We propose a new crossover operation and evaluate the performance and show the effectiveness of the operation by comparing with Greedy Subtour crossover (GSX) for solving ATSP.

## 1 Introduction

It is important to solve rapidly an optimal delivery route in the real world application. In this paper, we define the travelling cost as time. A delivery route planning is to minimize the total travelling cost.

When we solve the problem, we must consider one-way street (Fig.1). The cost A to B is not all the same the cost B to A, because the route A to B is not all the same the route B to A. Therefore, we must deal the problem with not symmetric TSP but an asymmetric TSP (ATSP). ATSP has asymmetric cost matrix, i.e.  $c_{ij} \neq c_{ji}$ , where  $c_{ij}$  is a time of the route  $i$  to  $j$ . It is not effective that we solve ATSP using optimization techniques of symmetric TSP whose cost  $c_{ij}$  equal to  $c_{ji}$ .

We intend to obtain an optimal or quasi-optimal route rapidly for practical use. We use Genetic Algorithm (GA) for fast ATSP solver. GA is one of the methods for search, learning and optimization. It is important to plan crossover operation for solving ATSP. Various crossover operations are proposed to solve symmetric TSP, but Few crossover operation have proposed for solving ATSP. y We focused on two crossover operations (methods for symmetric TSP) which are easy to preserve the character of the parent, because it is needed for crossover operation to

preserve the order of the cities which is appeared in the string of the parent for solving ATSP. We proposed a crossover operation for solving ATSP. The operation is based on one method for symmetric TSP, and confirmed the effectiveness of the proposed method by comparing with one method for symmetric TSP.

In this paper, we propose a new crossover operation for solving ATSP. The operation is based on the other method for symmetric TSP, and evaluate the effectiveness of proposed methods by comparing with the other method for symmetric TSP.

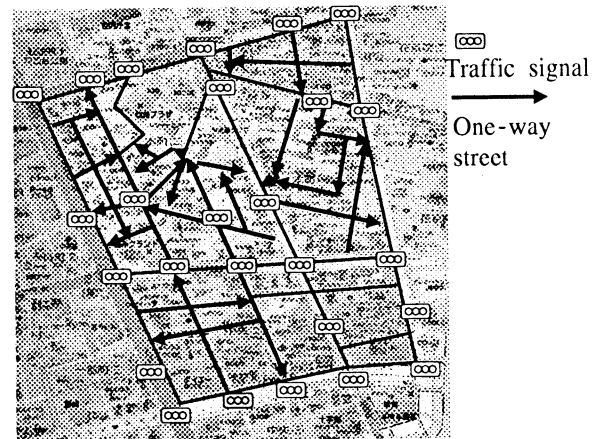


Fig.1: A map with traffic conditions

## 2 Delivery route planning

Delivery truck must go to all the delivery points. We must find the route which has minimum total travelling cost for effective delivery. Total travelling cost  $Z$  represents as follows:

$$Z = \sum_{i=1}^N c_{t_i, t_{i+1}} \quad (1)$$

where,  $(t_1, \dots, t_N)$  is an order of the delivery points,  $N$  is the number of cities,  $c_{t_i, t_{i+1}}$  is the travelling cost between the  $i$ th delivery point and  $i + 1$ th point, and  $t_{N+1}$  is equal to  $t_1$ .

In case we plan delivery route for a city area, we need to consider the waiting time for traffic signals[4] and difference between the time to turn right, turn left, and go straight. So we define  $c_{t_i, t_{i+1}}$  as follows:

$$c_{t_i, t_{i+1}} = T_{t_i, t_{i+1}} + C_{ss} + C_{sl} + C_{sr} + C_r + C_l + C_{stop} \quad (2)$$

$$T_{t_i, t_{i+1}} = \frac{d_{t_i, t_{i+1}}}{s} \quad (3)$$

$$C_{ss} = m_{ss} \times c_{ss} \quad (4)$$

$$C_{sl} = m_{sl} \times c_{sl} \quad (5)$$

$$C_l = m_l \times c_l \quad (6)$$

$$C_r = m_r \times c_r \quad (7)$$

$$C_{stop} = m_{stop} \times c_{stop} \quad (8)$$

where,  $d_{t_i, t_{i+1}}$  is the distance between the  $i$ th delivery point and  $i + 1$ th point,  $s$  is the mean speed,  $c$  is the time of the type of movement. For example,  $c_{ss}$  is a time to pass through the intersection with traffic signal.  $m$  is the number of each movement between the  $i$ th delivery point and  $i + 1$ th point. For example,  $m_{ss}$  represents the number of passing through intersection with traffic signal between  $i$ th delivery point and  $i + 1$ th point. Subscript represents the type of movement. "sl" and "sr" mean to turn left and right the intersection with traffic signal respectively. "l" and "r" mean to turn left and right the intersection without traffic signal respectively. "stop" means that we complete stop at the intersection.

### 3 Method for optimization

#### 3.1 Previous works

We focused on two crossover operation which are proposed for solving symmetric TSP because it is needed for crossover operation to preserve the order of the cities which is appeared in the string of the parent for solving ATSP. They are Edge recombination crossover (EX)[1] and Greedy Subtour crossover (GSX)[2].

EX is the crossover operation which is easy to preserve the character of the parent. The procedure of EX is as follows: We chose a city as the current city under the defined rule. We consider the four undirected edges incident to the current city in the parents and select an edge under the defined rule. If none of the

parental edges leads to an unvisited cities, create an edge to a randomly chosen unvisited city. repeat until all cities have been visited. The offspring can inherit the tour of the parent intermittently.

GSX is the crossover operation that acquires sequence of the tour of parents as long as possible. This crossover operation is designed for fast symmetric TSP solver, and have to be used with 2-opt method. The procedure of GSX as follows: First, a city is chosen at random. Next, pick up the cities from the parents alternately. The rest of cities are added to offspring randomly. if acquired so short sequence, there is a risk that the offspring is almost same as formed by random search.

We proposed the crossover operation based on EX (method1) for solving ATSP. Method1 is combined EX with the property of GSX which acquires sequence of the tour of the parent as long as possible. We defined the rule to select edge so as to inherit the order which is appeared in the parent as long as possible. It was shown that method1 is superior to EX in solving ATSP[3].

#### 3.2 Proposed method

We propose a new method (method2) for solving ATSP which is based on GSX. We improve the way of inheritance of GSX. The offspring can inherit the order of the cities which appears in the parent as long as possible.

The procedure of the method2 is as follows: First, a city is chosen at random. Next, pick up the cities from the parents alternately. The rest of cities are added to offspring not randomly but preserving the order in which they appear in the parent, skipping over all cities already present in the offspring. In this way, the offspring can inherit the longer sequence of the tour of the parent. Even if the tour of the parent break at once, a part of the tour is inherited to offspring.

An example is shown in Fig 2. Each individual is expressed path representation. Suppose that the chromosome of parents are [A,B,C,D,E,F,G,H] and [G,E,C,A,D,B,F,H]. The city C is chosen at random, Pick up the cities begin with A and next B. In the same way, the city D is added, but A does not because the city A is already appeared in offspring. Then, we add rest of the cities in the order which appears in the parent who transfer shorter tour to the offspring.

The procedure of the GA is as follows:

- genetic coding is path representation,
- the fitness is equal to the total travelling cost,

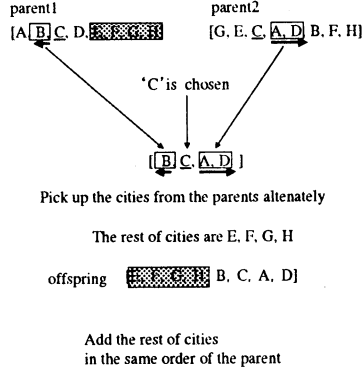


Fig.2: Crossover of the method2

- natural selection is ranking selection,
- mutation is 2-opt method,

At first, initial population is constructed randomly. By using selection by ranking strategy, higher ranked individuals can survive next generation and lower ranked individuals are replaced by offsprings. If the individuals which has the same chromosome are exist in the population, We eliminate one of them in order to preserve the variety of population. Mutation is executed by 2-opt method towards all individuals in a population. If the fitness of individual is not improved by 2-opt, that individual has no change.

## 4 Experiments

### 4.1 Benchmark test

We compare the result of the proposed crossover operations (method1, method2) with that of GSX by ATSP-benchmark-problems which are contained in TSPLIB [5]. Here, we evaluate the average of the total travelling cost and the convergence.

The GA parameters are given by the following.

- Population size  $N$  is 100,
- The number of generation to stop is 5000,

We solve three problems (38-city, 53-city, and 70-city). We execute the trial 20 times for each problem. The results are shown in Table1. The characteristic of convergence is shown Fig.3, Fig.4 and Fig.5.

Method2 converges the solution whose relative error is about less than 1% for each problem in early generation. Method1 converges slowly, but gives the best solution finally for each problem. GSX converges slowly. Average of the solutions of relative error is about more than 10% for each problem.

From the result, in order to solve the problem rapidly, We propose the way that we solve ATSP by using method2 at first, and switch method2 to method1 when the fitness of elite individual does not be improved for certain generations.

Table1 : The solutions for three problems

	method	mean	minimum	maximum
38city	method1	1559	1530	1598
	method2	1592	1546	1654
	GSX	1698	1559	1779
53city	method1	7003	6905	7213
	method2	7024	6905	7387
	GSX	8514	8014	9135
70city	method1	39357	38984	39950
	method2	39716	38900	40316
	GSX	44585	42306	46613

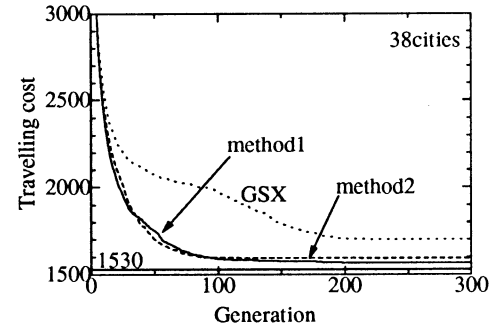


Fig.3: Convergence for 38city problem

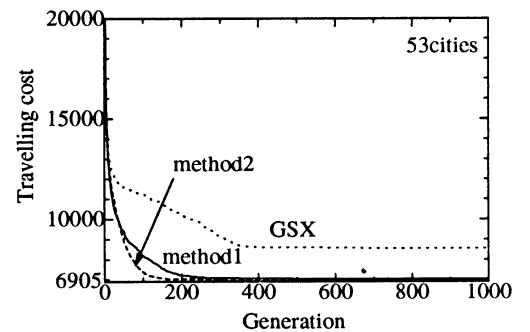


Fig.4: Convergence for 53city problem

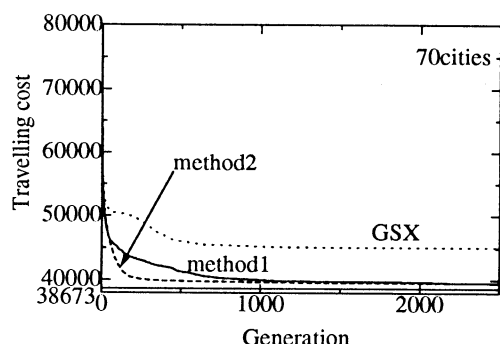


Fig.5: Convergence for 70city problem

## 4.2 An application

We deal with the delivery route optimization problem for a part of Sendai-city area (Figure 1). We use the equation (2) considering first five terms, and solve the problem using proposed methods.

We can get the route which is in Fig.6. As a result, the route which is given by method1 and the route which is given by method2 are the same route. It takes almost the same time to give the route. The reason for the result is the number of the delivery point is too small, 15 points.

We suppose that we get almost the same result as that of ATSP-benchmark-problem when we solve the problem which has large number of delivery points.

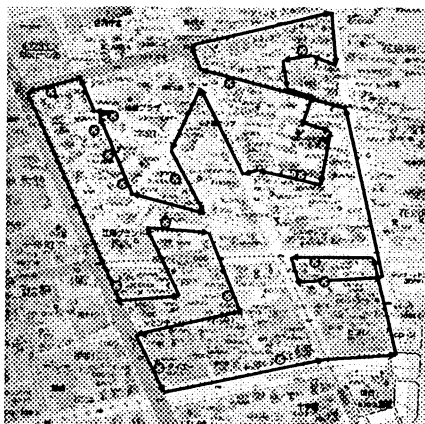


Fig.6: optimal delivery route

## 5 Conclusion

In this paper, We propose a new crossover operation -preseGSX and evaluate the performance of the

proposed methods by ATSP-benchmark-problem. We solve the optimal delivery route optimization problem and obtain the route which is shown in Fig.6.

As a result, proposed methods are superior to GSX. Method2 is superior to others limited in early generations, but Method1 gives the best solution finally.

It means that we solve ATSP by using method2 at first, and switch method2 to method1 when the fitness of elite individual does not be improved for certain generations.

In future works, we will decide the cost  $c_{t_i, t_j}$  and route between the delivery points automatically, and execute route planning with large number of delivery points.

## 6 Acknowledgements

This research is supported by East Japan Railway Company Endowed Chair of Large Scale System Stages Engineering in Tohoku University.

## References

- [1] L.Davis: *Genetic Algorithm Handbook*, Van Nostrand Reinhold, 1990.
- [2] Sengoku, H. Yoshihara, I.: "A Fast TSP Solution using Genetic Algorithm (in Japanese)" *Information Processing Society of Japan 46th Nat'l Conv.*, 8D-4, 2, p.305, 1993.
- [3] Takeda, A. Yamada, S. Sugawara, K. Yoshihara, I. Abe, K.: "ATSP solution using Genetic Algorithm (in Japanese)" *TOHOKU-section JOINT Conv record of IEEJ*, p.358, 1998.
- [4] Takeda, A. Yamada, S. Sugawara, K. Yoshihara, I. Abe, K.: "Optimization of delivery route planning in a city area using Genetic Algorithm (in Japanese)" *Information Processing Society of Japan 57th Nat'l Conv.*, Vol. 2, pp. 321-322, 1998.
- [5] Reinelt, G.: TSPLIB, <http://www.crpc.rice.edu/softlib/catalog/tsplib.html>

## Time Series Prediction Modeling by Genetic Programming without Inheritance of Model Parameters

M. Numata K. Sugawara S. Yamada I. Yoshihara K. Abe  
Graduate School of Engineering, Tohoku University  
Aoba05, Aza-Aramaki, Aoba-ku, Sendai, 980-8579 JAPAN  
{numata,sugawara,yamada,yoshiha}@largesys.ecei.tohoku.ac.jp  
abe@abe.ecei.tohoku.ac.jp

### Abstract

We try automatically to build time series prediction model by Genetic Programming(GP). The method consists of two stages. The first stage is to determine composite function model by GP, the second one is to optimize parameters included in the model by the back propagation method. We experiment time series prediction of chaotic time series and wind velocity as application examples.

## 1 Introduction

Time series prediction plays an important role in various fields, *e.g.* business, industry and engineering. However, since future events involve uncertainty, the predictions are usually not perfect[2]. Therefore, when we predict time series, we need to build an appropriate prediction model of time series. The objective of our work is to form the prediction model which has few prediction errors easily and automatically.

The time series prediction is to map points from lag space(input parameters of the model,  $x_{t-1}, x_{t-2}, \dots, x_{t-\tau}$ , *i.e.* current input and  $\tau$ ) to an estimate of the future value, that is,

$$\hat{x}(t) = f(x(t-\tau), \dots, x(t-2), x(t-1)) \quad (1)$$

where  $\tau$  denotes the time span used for prediction,  $t$  denotes current time. We propose a method of building prediction model, that consists of two stages: one is determining the functional form of the model and the other is determining parameters of the model.

In the former stage we utilize GP, because GP has succeeded in the field of automatic define function[3][4][5][6]. In the latter stage we utilize the back propagation method[7] which is famous for a learning method of neural network.

In the previous work, we generated the mathematical prediction model of sun-spots time series using

GP, which was composed of only the four arithmetic operations and a saturated function[1]. The prediction result was satisfactory on the whole.

In this paper, we propose GP utilizing many operations besides the four arithmetic operations and a saturated function in order to aim at improving the prediction and attempt to apply to wind velocity as actual data.

## 2 Modeling by GP

### 2.1 Model Building

GP has many advantages to build a prediction model, for example to compose functions is easy, special knowledge is not required to find a solution and it can cope with the change of various condition.

GP handles tree-structured chromosome that represents composite function model(Fig.1).

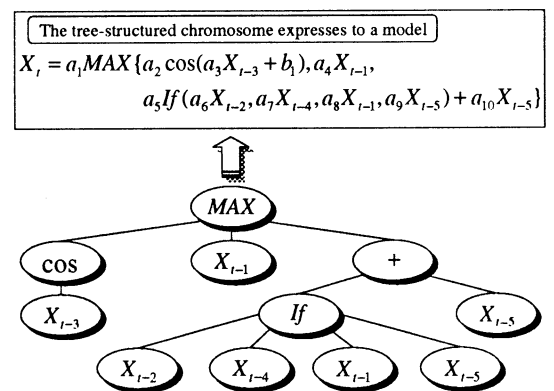


Figure 1: Chromosome of Tree-Structure

Nodes involved in the tree-structured chromosome are divided into two categories: one represents input

data(leaves), the other represents operators. Table 1 presents operators involved in the model.

The tree-structure of Fig. 1 expresses the following mathematical model,

$$X_t = a_1 MAX\{a_2 cos(a_3 X_{t-3} + b_1), a_4 X_{t-1}, a_5 If(a_6 X_{t-2}, a_7 X_{t-4}, a_8 X_{t-1}, a_9 X_{t-5}) + a_{10} X_{t-5}\} \quad (2)$$

$$= f(X_1, X_2, \dots, X_n, a_1, a_2, \dots, a_m) \quad (3)$$

where coefficients  $a_1, a_2, \dots, a_m$  are model parameters, whose values are not given from the parents, but determined by the back propagation method.

Table 1: Operators in the Model

Operation	No of Args	Definition
$Sin(x_1)$	1	$\sin(a_1 x_1 + b)$
$Cos(x_1)$	1	$\cos(a_1 x_1 + b)$
$Atan(x_1)$	1	$\frac{1}{2}(1 + \frac{1}{\pi} atan(a_1 x_1 + b))$
$Log(x_1)$	1	$\log a_1 x_1 + b $
$Sig(x_1)$	1	$1/(1 + e^{-(a_1 x_1 + b)})$
$Sinh(x_1)$	1	$\sinh(a_1 x_1 + b)$
$Cosh(x_1)$	1	$\cosh(a_1 x_1 + b)$
$Abs(x_1)$	1	$ x_1 $
$Exp(x_1)$	1	$e^{a_1 x_1 + b}$
$Unit(x_1)$	1	$\begin{cases} x_1 &  x_1  \leq \frac{1}{2} \\ 0 &  x_1  > \frac{1}{2} \end{cases}$
$Nop(x_1)$	1	$x_1$
$Add(x_1, x_2)$	2	$a_1 x_1 + a_2 x_2$
$Mul(x_1, x_2)$	2	$a_1 x_1 x_2$
$Div(x_1, x_2)$	2	$\begin{cases} a_1 \frac{x_1}{x_2} & x_2 \neq 0 \\ a_1 x_1 & x_2 = 0 \end{cases}$
$MAX(x_1, x_2, x_3)$	3	the largest variable
$MID(x_1, x_2, x_3)$	3	the middle variable
$MIN(x_1, x_2, x_3)$	3	the smallest variable
$if(x_1, x_2, x_3, x_4)$	4	$\begin{cases} a_1 x_3 & x_1 \leq x_2 \\ a_2 x_4 & x_1 > x_2 \end{cases}$

## 2.2 Procedure of GP

The procedure of modeling the system by GP is as follows.

- Initializing Population

Individuals of the first generation are generated at random as a group.

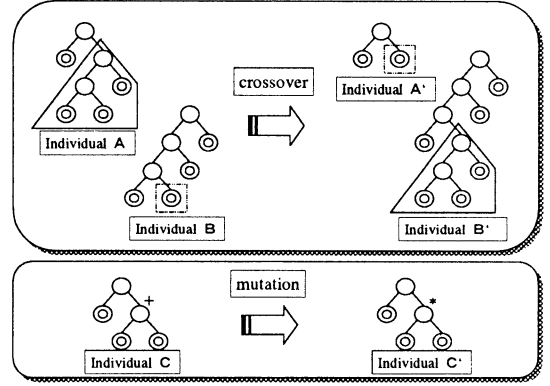


Figure 2: Crossover and Mutation

- Genetic Operations

- Crossover

A pair of individuals are chosen at random as parents. A part of chromosome is cut out from the parent. The segments of chromosome are exchanged each other and a new individual is born as a child (Fig. 2). The functional form of the model is inherited, but parameters in the model are not. Parameters are determined by the back propagation method whenever a child is generated.

- Fitness

Fitness is determined by prediction errors and simplicity of model.

$$Fitness = \frac{1}{E} \quad (4)$$

$$E = \frac{1}{2} \sum_{i=1}^k (\tilde{y}_i - y_i)^2 + h(m) \quad (5)$$

$$h(m) = cm \quad (6)$$

$$\left( \begin{array}{l} c : constant \\ m : number\ of\ nodes \end{array} \right)$$

The first term of the right side of equation (5) means error squares of the prediction. The second term is so called MDL[8] and means complexity of the prediction model. The basic idea of MDL is “the simpler, the better”. MDL terms control the model to become gigantic.

- Natural Selection

Fitness of individual is determined mainly by prediction errors. Next generation indi-

viduals are selected by the ranking. An individual of higher fitness is easy to survive more than an individual of lower fitness.

– Mutation

One of the inner nodes of the tree is chosen at random and the operator is changed.

• End of Search

If fitness of an elite reaches a certain value that we decide in advance, the search finishes. Even if it doesn't satisfy the value, searching will be stopped by predefined repetition time. In that case the elite of the last generation is regarded as the most appropriate model.

### 3 Experiments of prediction

#### 3.1 Chaotic Time Series

The first example is to predict Lorenz attractor model. Lorenz attractor model is generated by differential equations systems.

$$\frac{dx}{dt} = -10x + 10y \quad (7)$$

$$\frac{dy}{dt} = 28x - y - xz \quad (8)$$

$$\frac{dz}{dt} = -\frac{8}{3}z + xy \quad (9)$$

These equations are nonlinear, so it is difficult to get the analytic solution. A trajectory from an initial point is derived using Runge-Kutta method. We generated 250 data of  $z(t)$  as time series. The first 150 data are used for parameter fitting and the last 100 data for validation.

We used three methods for generating the prediction model, neural network(NN), auto regressive(AR) and proposed GP, and then we get the following prediction Fig. 3, 4 and 5.

Table 2 shows the gross square errors, the greatest errors and the average of errors by three models, *i.e.* NN model, AR model and proposed GP.

Predicted peaks of NN and AR are apt to overshoot(Fig. 3 and 4). It is difficult for these methods to adjust the peak of time series. Proposed GP succeeded in coping with peak of time series(Fig. 5). From the results, we can see that the model by proposed GP can predict most skillfully.

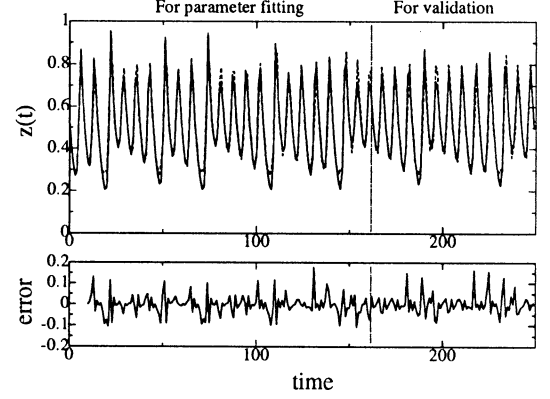


Figure 3: Prediction by Neural Network

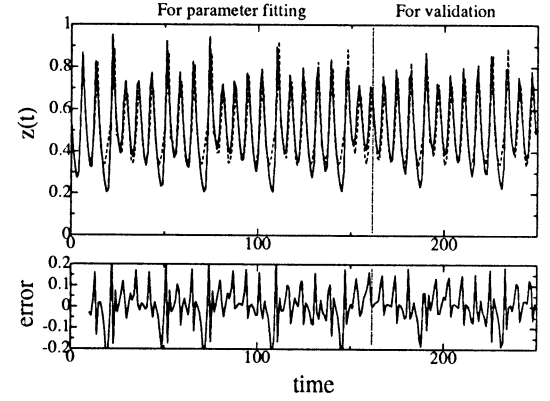


Figure 4: Prediction by Auto Regressive

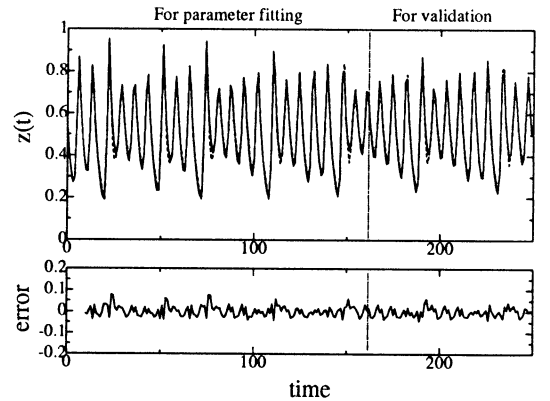


Figure 5: Prediction by Proposed GP

Table 2: Prediction Results (by 3 models)

	neural network	auto regressive	proposed GP
the gross square error	0.5435	2.0440	0.1389
the greatest absolute error	0.1767	0.2539	0.0781
the average of absolute error	0.0333	0.0684	0.0198

### 3.2 Prediction of Wind Velocity

The second example is to predict of wind velocity of actual data. The wind velocity is measured at Kumagawa every 30 seconds for 24 hours. The prediction result by proposed GP is Fig. 6.

It is difficult to predict actual data like wind velocity, but the absolute prediction error average of the model by GP was limited within 5% of wind velocity.

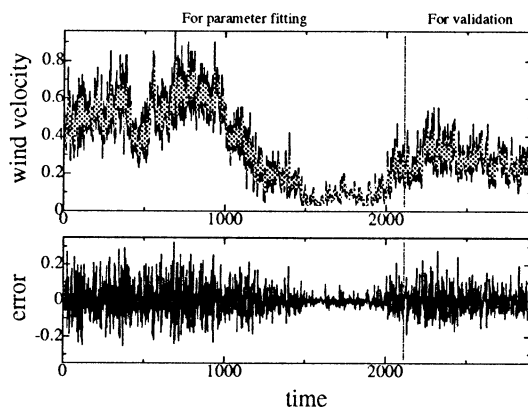


Figure 6: Wind Velocity Prediction by Proposed GP

## 4 Conclusion

A model building method using GP is proposed for time series prediction. The method automatically produces composite function model, which are applicable to many a complicated time series, *e.g.* chaotic time series, sun-spots and wind velocity.

The method is easy to compose function models, because no special knowledge is required. It can sometimes produce very tricky model that analysts and engineers scarcely think out.

In our future works, we will investigate influences on prediction with and without inheritance of model parameters in GP.

## Acknowledgements

The authors wish to thank the East Japan Railway Company Endowed Chair of Large Scale System Stages Engineering in Tohoku University for supporting their research.

## References

- [1] Numata, M., Sugawara, K., Yamada, S., Yoshihara, I. and Abe, K., "Time Series Prediction by Genetic Programming", Late Breaking Papers, GP-98, pp.176-179, (1998).
- [2] Bovas, A. and Johannes, L., "Statistical Methods for Forecasting", John Wiley & Sons, 1983.
- [3] Koza, J., "Genetic Programming", On the Programming of Computers by means of Natural Selection, MIT Press, 1992.
- [4] Yoshihara, I., Sato, S., "Nonlinear Model Building Method with GA and GMDH", IPSJ, AI-105, pp.1-6, (1996) (in Japanese).
- [5] Oakley, H., "Two Scientific Applications of Genetic Programming", Stack Filters and Non-Linear Equation Fitting to Chaotic Data, in Advances in Genetic Programming, (ed. Kenneth E. Kinnear, Jr.), MIT Press, 1994.
- [6] Iba, H., deGaris, H. and Sato, T., "Recombination Guidance for Numerical Genetic Programming", in Proc. of 2nd International Conference on Evolutionary Computation, IEEE Press, 1995.
- [7] Rumelhart, D., Hinton, G. and Williams, R., "Learning representations by back-propagation errors", Nature 323, pp.533-536, (1986).
- [8] Rissanen, J., "Stochastic Complexity", J. of Royal Statist. Soc. B, vol. 49, no. 3, pp. 223-239 and pp. 252-265, (1987).



# Information Transformation by Virus-Evolutionary Genetic Programming

Naoyuki KUBOTA\*, Fumio KOJIMA\*, Setsuo HASHIMOTO\*, and Toshio FUKUDA\*\*

\* Dept. of Mechanical Engineering

Osaka Institute of Technology

5-16-1 Omiya, Asahi-ku, Osaka 535-8585, Japan

\*\* Center for Cooperative Research in Advanced Science

and Technology, Nagoya University

1 Furo-cho, Chikusa-ku, Nagoya 464-8603, Japan

**Abstract** - This paper deals with a genetic programming (GP) for information translation. The GP can generate a structured computer program, but it is difficult to define recursive functions automatically. Therefore, this paper proposes a virus-evolutionary genetic programming (VE-GP) composed of two population; host and virus. Here a virus plays the role of automatic function definition. First, the VE-GP is applied to a function approximation problem and the simulation result shows that the VE-GP can generate a function to approximate the given function with small errors. Next, the VE-GP is applied to the information transformation for a classification task and the simulation result shows that the VE-GP can generate a function to classify a given data set.

**Key words:** Computational Intelligent, Genetic Programming, Classifier System

## I. INTRODUCTION

Machine intelligence has been discussed in various fields such as artificial intelligence, cognitive science, computer science, knowledge engineering, and intelligent robotics<sup>1-10</sup>. Artificial intelligence (AI) aims to describe and build an intelligent agent which perceives its environment, makes decisions, and takes actions<sup>1</sup>. Recently, computational intelligence including fuzzy computing, neural computing, and evolutionary computing, has been successfully used as an intelligent technique based on the concept of learning, adaptation, and evolution<sup>2-4</sup>. High intelligence emerges from the close linkage between perception, decision making, and action. Therefore, we have proposed the concept of structured intelligence based on the close linkage, and applied it to various robotic systems<sup>12</sup>. However, it is very difficult to realize high perception capabilities on the robotic systems. Though the image processing can extract significant features, there is a restriction of on-line computational cost. Consequently, the intelligent agent should survive in its dynamic environment by processing sensed quantitative information into meaningful qualitative information for itself. To realize the information transformation, we apply a genetic programming and fuzzy inference system.

Koza has proposed genetic programming (GP) to evolve Lisp programs to accomplish various tasks<sup>9,10</sup>. The GP can generate a structured computer program, but it is difficult to define recursive functions automatically. Therefore, this paper proposes virus-evolutionary genetic programming (VE-GP) which is a coevolutionary optimization algorithm based on mutualism. The VE-

GP is applied to two types of computer simulations; (a) a function approximation task and, (b) information transformation for a classification task. We discuss the effectiveness of the VE-GP through the computer simulation results.

## II. COEVOLUTIONARY COMPUTING

Coevolutionary computation (CEC) has been applied to much complicated problems<sup>13-17</sup>. CEC is generally composed of several species with different types of individuals (candidate solutions), while standard EC has a single population of individuals. In the CEC, crossover and mutation are performed only in a single species, because a species is used as a group of interbreeding individual, not normally able to interbreed with other such groups. The selection can be performed among individuals in a species and among species. The concept of coevolution is based on two basic interactions: cooperation and competition. These interactions are generally determined by the benefit and harm between several species. In general, there are various interactions in two or more species. These interactions are dependent on the influence of a species against the other. To simplify the interaction, we consider only two species: *A* and *B*. Table 1 shows the interaction between two species.

The concept of coevolution has been applied to various types of design problems and pattern classification problems. In design problems, the host-parasite model is often used<sup>13-17</sup>. The host species tries to optimize the design parameter to the simulation parameters (environmental conditions) given by the parasite species, while the host species tries to minimize

Table 1 Interaction between two species A and B

		Influence of B to A		
		+ (benefit)	0	- (harm)
Influence of A to B	+	mutualism	commensalism	parasitism
	0	commensalism	neutralism	amensalism
	-	parasitism	amensalism	competition

the performance of the design parameters. This is similar to a population which tries to adapt to its dynamically changing environment. Furthermore, this model is applied to the learning of NNs and FS. The host and parasite species are a set of NNs and a set of learning data. The parasite selects and provides the learning data difficult for NNs to learn.

In the classification problems, the input data are directly used for the classification, or the input data are translated into qualitative information by human operators. This translation is a very difficult task and it takes much effort. Therefore, this task is also introduced into the optimization process of classifier system<sup>17</sup>. Generally, the classifier system is organized as follows;

- step 1: preprocessing of input data,
- step 2: classification by classifier system, and
- step 3: post-processing of output data.

The preprocessing includes feature extraction and feature selection. To build a well performed classifier, preprocessing is very important, because the translated information differentiates a class from other classes. By using computational intelligent methods, we can develop the following systems (Fig.1):

- (1) GP + NN (FS),
- (2) GA + NN (FS),
- (3) GP (GA) + NN (FS) + GP, etc.

In the case (1), the GP plays the role of feature extraction, *i.e.*, the GP translates a set of given raw data into meaningful data for the classifier (NN or FS). In the case (2), the GA plays the role of feature selection, *i.e.*, the GA reduces input dimension to the classifier (NN, or FS). In the case (3), the last GP plays the role of post-processing. In this way, the coevolution (co-optimization) of GP (GA) and FS (NN) can generate high intelligent systems.

### III VIRUS-EVOLUTIONARY GENETIC PROGRAMMING

This section proposes a genetic programming based on virus theory of evolution<sup>13</sup>. The VE-GP is composed of two population; host and virus. The host population means a set of candidate solutions which represent

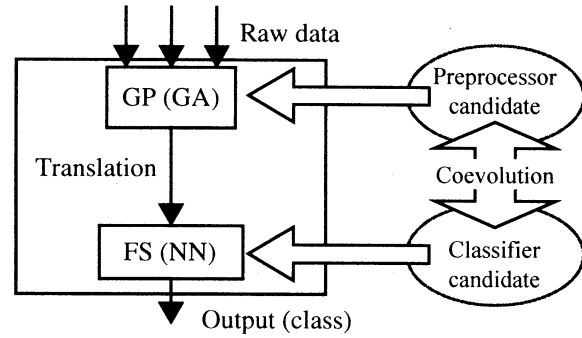


Fig.1 Fuzzy inference system (NN) with GP (GA)

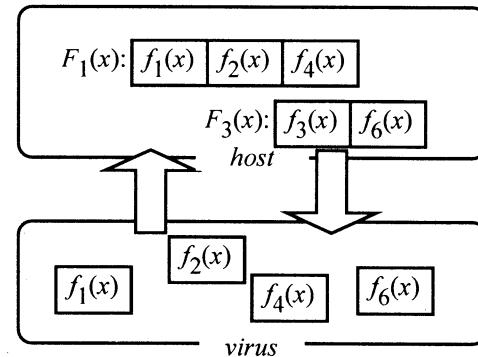


Fig.2 Virus-evolutionary genetic programming; VE-GP

computer programs for translating (Fig.2). The virus population plays the role of automatic function definition through the interaction with the host population. Consequently, a virus is represented as a recursive function used in the host population. The VE-GP has two virus infection operators; reverse transcription and incorporation. A virus overwrites its substring on the string of a host by the reverse transcription to generate new candidate solutions, and takes out a substring from an infected host by the incorporation to generate new recursive functions. Figure 3 shows the procedure of virus infection mechanism. The fitness of a virus is calculated by using the degree of improvement by the infection to host population and the frequency used in the host population, *i.e.*, a good virus can be frequently used in a computer program for translating, and furthermore can improve the translation performance. Each virus has a measure of the strength of the virus infection (*fitvirus*). We assume that *fithost<sub>j</sub>* and *fithost<sub>j'</sub>* are fitness values of a

host  $j$  before and after the infection, respectively. The  $fitvirus_{i,j}$  denotes difference between  $fit_{host_j}$  and  $fit_{host_j}'$ , which is equal to the improvement value obtained by infecting to the host individual:

$$fitvirus_{i,j} = fit_{host_j}' - fit_{host_j} \quad (1)$$

$$fitvirus_i = \sum_{j \in S} fitvirus_{i,j} \quad (2)$$

where  $i$  is the virus number and  $S$  is a set of the host individuals which are infected by the virus  $i$ . In addition, the infection rate ( $infrate_i$ ) is updated according to the above improvement value.

$$infrate_{i,t+1} = \begin{cases} \alpha^{-1} \cdot infrate_{i,t} & \text{if } fitvirus_i > 0 \\ \alpha \cdot infrate_{i,t} & \text{otherwise} \end{cases} \quad (3)$$

where  $\alpha$  is a coefficient greater than zero and  $t$  is generation number. The increase of  $infrate_i$  accelerates the increase of effective functions by virus infection. Furthermore, each virus has a life force through generations as follows:

$$life_{i,t+1} = r \times life_{i,t} + fitvirus_i \quad (4)$$

where  $r$  is the life reduction rate, respectively. This life force plays the role of evaluating a function of virus to the host species. Paredis also emphasized the importance of lifetime fitness evaluation<sup>16</sup>. The basic architecture of the VE-GP is based on a steady-state model<sup>11</sup>. The steady-state GA (SSGA) basically exchanges the worst individuals with a pair of individuals generated by crossover and mutation<sup>11</sup>.

In the VE-GP, we apply a subtree crossover and three mutation operators; (a) homogeneous replacement, (b) terminal replacement, and (c) inserting mutation. The fitness of a host is calculated by using the string length and performance index. As the selection mechanism, we use steady-state model which replace the worst host with a child generated by the crossover and mutation.

### III. COMPUTER SIMULATION

This section shows two simulation result of the VE-GP. First, we apply the VE-GP to an approximation task of the following function,

$$y = x \sin(4\pi x) + \exp(-x) \sin x + 3 \quad x=0.01, 0.02, \dots, 1.0 \quad (5)$$

The host and virus population sizes are 500 and 100, respectively. The crossover probability is 0.6. An initial candidate solution (host) is generated by repeating the

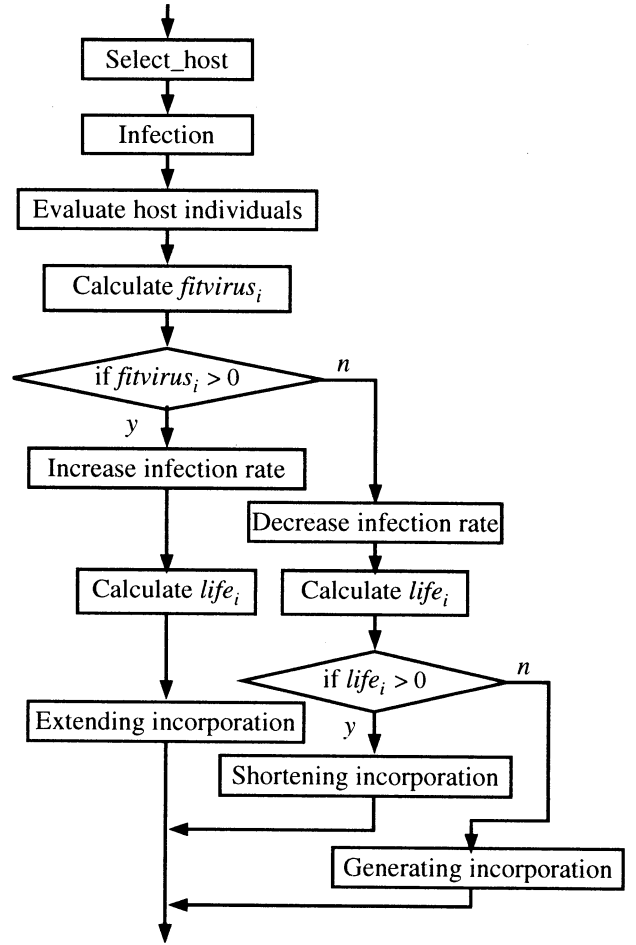


Fig.3 Procedure of virus infection

inserting mutation several times. The evaluation times is 500000. The genotype is as follows:

Unary operator :  $\sin, \cos$

Binary operator :  $+, -, *, \min, \max$

Terminals :  $x(t), x(t-1), x(t-2), \text{constants}$

The number of given points is 100. The performance index is defined as a square error. Figure 4 shows a simulation result. Though the genotype does not include the exponential operator, the VE-GP can generate a function approximating a given function.

Next, we apply the VE-GP to information transformation for a classification task. A classification task is often difficult to solve if the structure of a given data set is much complicated. Therefore, we use the VE-GP as a preprocessing system for the classification task. The data set for the classification task is generated by the curve of eq.(5). The upper side of the curve is class 2, and the other side is 1. Figure 5 shows a teaching data set generated by the curve and the number of data is 100. The first 50 points are located near the curve and the other points are located randomly. Figure 6 shows a simulation result transformed by a generated function.

The simulation result shows that the given data set can be transformed into an other data set easy to classify. In this case, the classification can be done by a straight line in the figure.

## VI. CONCLUSION

This paper proposed a new genetic programming based on the virus theory of evolution (VE-GP), which consists of host and virus populations. The virus population plays the role of automatic function definition. Next, we applied the VE-GP to a function approximation task and the information transformation for a classification task. The VE-GP can generate functions accomplishing various given tasks.

As a future subject, we must discuss the mathematical analysis of the VE-GP. Furthermore, we intend to apply the VE-GP with fuzzy inference system to a perception mechanism of a mobile robotic system.

## References

1. S.J.Russell and P.Norvig (1995), *Artificial Intelligence*, Prentice-Hall, Inc.
2. J. M. Zurada, R. J. Marks II, and C. J. Robinson (1994), *Computational Intelligence - Imitating Life*, IEEE Press.
3. M.Palaniswami, Y.Attikiouzel, R.J.Marks II, D.Fogel, and T.Fukuda (1995), *Computational Intelligence - A Dynamic System Perspective*, IEEE Press.
4. J.-S.R.Jang, C.-T.Sun, and E.Mizutani (1997), *Neuro-Fuzzy and Soft Computing*, Prentice-Hall, Inc.
5. C.G.Langton (1995), *Artificial Life -An Overview*, The MIT Press.
6. D.B.Fogel (1995), *Evolutionary Computation*, IEEE Press.
7. J.Holland (1975), *Adaptation in Natural and Artificial Systems*, Ann Arbor:University of Michigan Press.
8. D.E.Goldberg (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Welsey.
9. J.Koza (1992), *Genetic Programming*, Massachusetts: The MIT Press.
10. J.Koza (1994), *Genetic Programming II*, Massachusetts: The MIT Press.
11. G.Syswerda (1991), "A Study of Reproduction in Generational and Steady-State Genetic Algorithms." In *Foundations of Genetic Algorithms*, San Mateo: Morgan Kaufmann Publishers, Inc.
12. N.Kubota and T.Fukuda (1998), Sensory Network for Mobile Robotic Systems with Structured Intelligence, *Journal of Robotics and Mechatronics*, Vol.10, No.4, pp.338-349.
13. N.Kubota, K.Shimojima, and T.Fukuda (1996), "The Role of Virus Infection in Virus-Evolutionary Genetic Algorithm", *Journal of Applied Mathematics and Computer Science*, Vol.6, No.3, pp.415-429.
14. W.D.Hillis (1991), "Co-Evolving Parasites Improve Simulated Evolution as An Optimization Procedure", *Artificial Life II*, edited by C.G.Langton, C.Taylor, J.D.Farmer and S.Rasmussen, Addison Wesley, pp. 313-324.
15. P.J.Angeline and J.B.Pollack (1993), "Competitive Environments Evolve Better Solutions for Complex Tasks", *Proc. of The Fifth International Conference on Genetic Algorithms*, pp.264-270.
16. J.Paredis (1995), "Coevolutionary Computation", *Artificial Life*, Vol.2, No.4, pp.355-375.
17. J..R.Sherrah, R.E.Bogner, A.Bouzerdoum (1997), The Evolutionary Pre-Processor, *Proc. of Second Annual Conderence on Genetic Programming*, pp.305-312.

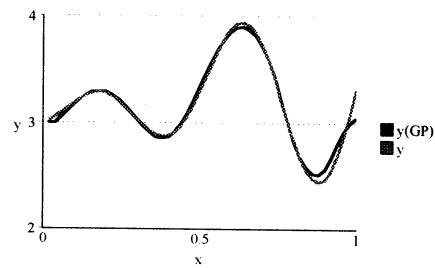


Fig.4 A simulation result of function approximation

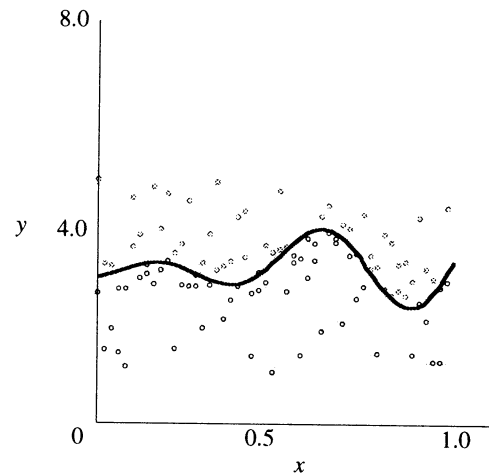


Fig.5 Teaching data for classification

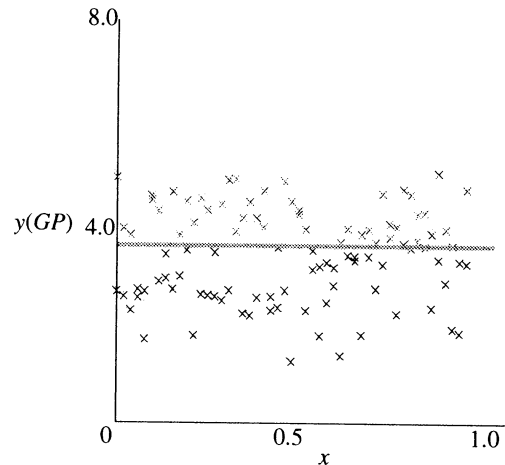


Fig.6 A simulation result of classification by VE-GP

## A Description of Dynamic Behavior of Sensory/Motor Systems with Fuzzy Symbolic Dynamic Systems

I. Takeuchi    T. Furuhashi

Department of Information Electronics  
Nagoya University Graduate School of Engineering  
Furo-cho, Chikusa-ku, Nagoya. 464-8603

### Abstract

This paper presents a model of continuous sensory/motor systems for autonomous agents in navigation problems. Markov environmental model and sequential plan are extended with fuzzy sets, which present the mathematical transformation from discrete state space to continuous state space. The extended fuzzy environmental model and fuzzy sequential knowledge enable the identification of the continuous sensory/motor systems with gradient descent-based parameter estimation algorithm. A simulation demonstrates the feasibility of the proposed method.

## 1 Introduction

This paper deals with a navigation problem of autonomous agents. The problem is to design a continuous sensory/motor system of the agent such that it can autonomously navigate in the environment to fulfill required tasks with minimum costs.

In the beginning of the problem formulation, the designer has to make the environmental model  $M_X(\mathbf{X}, C_X)$ , where  $\mathbf{X} \subseteq \mathbf{R}^N$  denotes the state space of the environmental model, and  $C_X(\mathbf{x}, \dot{\mathbf{x}})$ ,  $\mathbf{x} \in \mathbf{X}$  indicates the costs<sup>1</sup> of the navigation in the environment. The sensory/motor system is a dynamic system expressed by the following equations:

$$\dot{\mathbf{x}}_t = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t) \quad \mathbf{y}_t = \mathbf{g}(\mathbf{x}_t, \mathbf{u}_t) \quad (1)$$

where  $\mathbf{u}_t = (u_{1,t}, \dots, u_{P,t})$  and  $\mathbf{y}_t = (y_{1,t}, \dots, y_{R,t})$  denote the sensory inputs and motor outputs at time  $t$ , respectively, the function  $\mathbf{f}$ , called state equation, describes the dynamic characteristics, and  $\mathbf{g}$ , called output equation, gives the input/output characteristics. When the designer can define the environmental model and the sensory/motor system in the forms described above, it is possible<sup>2</sup> to find the optimal state

transition series  $\mathbf{x}_1^{t_f*} = (\mathbf{x}_1^*, \mathbf{x}_2^*, \dots, \mathbf{x}_{t_f}^*)$ , which minimizes the total costs

$$J[\mathbf{x}_1^{t_f}] = \int_1^{t_f} C_X(\mathbf{x}_t, \dot{\mathbf{x}}_t) dt. \quad (2)$$

It is also possible to find the functions  $\mathbf{f}$  and  $\mathbf{g}$  which realize the optimal state transition series  $\mathbf{x}_1^{t_f*}$ .

However, it is almost impossible for the designer to set the continuous environmental model  $M_X(\mathbf{X}, C_X)$ . Generally, a Markov environmental model  $M_S(\mathbf{S}, C_S)$  substitutes for  $M_X$ .  $M_S(\mathbf{S}, C_S)$  has discrete state space  $\mathbf{S}$  and the costs  $C_S\{s, \Delta s\}$ ,  $s \in \mathbf{S}$ . When the designer sets a Markov model  $M_S(\mathbf{S}, C_S)$ , it is possible<sup>3</sup> to find the optimal discrete state transition sequence  $\mathbf{s}_1^{\tau_f*} = (s_1^*, s_2^*, \dots, s_{\tau_f}^*)$ , which minimizes the total costs

$$J[\mathbf{s}_1^{\tau_f}] = \sum_{\tau_1}^{\tau_f-1} C_S(s_\tau, \Delta s_\tau). \quad (3)$$

But the discrete state space  $\mathbf{S}$  given by the top-down approach cannot be used for that of continuous sensory/motor system. This problem is conceptualized as 'a symbol grounding problem[1]' or 'a state-action deviation problem[2]'. Several attempts[2][3] have been made for the problem with the approach of self-organization of the state space. Although they have been paid attention from the viewpoint of artificial life or cognitive science, they have only realized poor performances.

This paper approaches to this problem from the different angle. We pay attention to the "fuzziness" in the designer's knowledge about the environment. With the implication from fuzzy modeling, we shall represent the fuzziness in  $M_S$  and  $\mathbf{s}_1^{t_f}$  explicitly, which follows the fuzzy environmental model  $\tilde{M}_S$  and fuzzy sequential knowledge  $\tilde{\mathbf{s}}_1^{t_f}$ . The fuzziness in  $\tilde{M}_S$  or  $\tilde{\mathbf{s}}_1^{t_f}$  is defined with fuzzy sets and quantified with the mem-

<sup>1</sup>The required tasks can be also included in the costs  $C_X$ .

<sup>2</sup>We can find  $\mathbf{x}_1^{t_f*}$  with variation method.

<sup>3</sup>We can derive  $\mathbf{s}_1^{\tau_f*}$  with dynamic programming.

bership functions. This idea presents the mathematical transformation from discrete state space to continuous state space and gives the discussion for the continuous sensory/motor system.

In the following discussion, let us suppose that the Markov environmental model  $M_S(\mathcal{S}, C_S)$  and obtained sequential knowledge  $s_1^T$  were already given, and we could use the sample sensory/motor flows when the designer maneuvered the agent in the environment<sup>4</sup>.

## 2 Fuzzy Model for Dynamic Systems

### 2.1 Fuzzy Environmental Model

In the beginning let us suppose that the sensory inputs and motor outputs are also defined at each discrete state in a Markov environmental model. When sensory inputs and motor outputs at step  $\tau$  are denoted as  $v_\tau$  and  $\psi_\tau$ , respectively, the discrete sensory/motor system is given by the following equations:

$$\xi_{\tau+1} = \phi_{\xi_\tau}(v_\tau) \quad \psi_\tau = \gamma_{\xi_\tau}(v_\tau). \quad (4)$$

where  $\phi_\xi$  denotes the state transition function at state  $\xi$ , and  $\gamma_\xi$  indicates the function which defines the process at state  $\xi$ .

Next let us consider fuzzy state  $\tilde{\xi}$ . It means that the state of the system at time  $t$  are represented by a set  $\mathbf{x}_t = \{x_{1,t}, x_{2,t}, \dots, x_{N,t}\}$ , each of the element denotes the membership grade of the discrete state  $\xi_1, \xi_2, \dots, \xi_N$ , respectively. This fuzzification changes eqns.(4) into the following ones:

$$\dot{\mathbf{x}}_t = \sum_{i=1}^N x_{i,t} \mathbf{f}_i\{\mathbf{u}_t\} \quad \mathbf{y}_t = \sum_{i=1}^N x_{i,t} \mathbf{g}_i\{\mathbf{u}_t\}, \quad (5)$$

where  $\mathbf{F} = (\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_N)$  denotes the state equation and  $\mathbf{G} = (\mathbf{g}_1, \mathbf{g}_2, \dots, \mathbf{g}_N)$  denotes the output equation, which are corresponding to eqns.(1), respectively. Fig.1 shows the basic concept of fuzzy state  $\tilde{\xi}$ . The discrete state transition  $\xi_1 \rightarrow \xi_2 \rightarrow \xi_3$  are replaced with the fuzzy state transition[4]  $\tilde{\xi}_1 \rightarrow \tilde{\xi}_2 \rightarrow \tilde{\xi}_3$ .

The architecture of the sensory/motor system given by eqns.(5) is described as the mixture of submodels[5]. The  $i$ -th submodel, the state equation  $\mathbf{f}_i$  and the output equation  $\mathbf{g}_i$ , is mixed with the weight  $x_{i,t}$ . Fig.2 shows the block diagram of the system in the case of  $N = 2$ .

This fuzzification implies to introduce topological nature into the Markov environmental model. The membership vector  $\mathbf{x}_t = (x_{1,t}, x_{2,t}, \dots, x_{N,t})$  moves on the surface of convex polyhedron  $\mathbf{P}$ , which is given by a set of extremal points  $\{(1, 0, \dots, 0), (0, 1, \dots, 0), \dots, (0, 0, \dots, 1)\}$  in  $[0, 1]^N$ <sup>5</sup>. And state equation  $\mathbf{F}$  defines a vector field  $\mathbf{P} \rightarrow \mathbf{P}$ . Fig.3 illustrates an example of the movement of  $\mathbf{x}_t$  on  $\mathbf{P}$ .

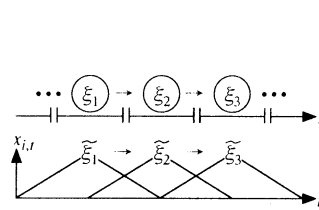


Fig.1 Concept of Fuzzy State

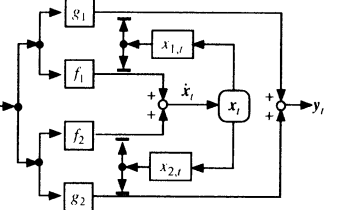


Fig.2 Block Diagram

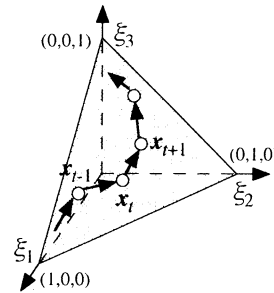


Fig.3  $\mathbf{x}_t$ 's Movement on  $\mathbf{P}$

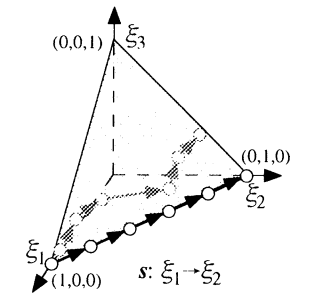


Fig.4 Sequential Knowledge

### 2.2 Fuzzy Sequential Knowledge

As mentioned in the previous subsection, fuzzification of discrete states brings the sequential knowledge to the discussion on topological space, convex polyhedron  $\mathbf{P}$ . Let us consider the case where the sequential knowledge is given as  $s : \xi_1 \rightarrow \xi_2$ . On the convex polyhedron  $\mathbf{P}$ , it is supposed that  $s : \xi_1 \rightarrow \xi_2$  is described as the uniform linear movement on the line-segment  $\xi_1 \xi_2$  as denoted by dark arrows shown in fig.4. In the same way, it is assumed that fuzzy sequential knowledge  $\tilde{s} : \xi_1 \rightarrow \xi_2$  is expressed as the movement "around" the line segment  $\xi_1 \xi_2$  at "about" uniformly as denoted by light arrows shown in fig.4.

Next, we shall confine our attention to define the fuzziness, "around" and "about". A membership function  $\mu_{\text{around}}$  as shown in fig.5(a) quantifies the fuzziness "around".  $\mu_{\text{around}}$  evaluates the perpendicular deviation from the line segment  $\xi_1 \xi_2$  as shown in fig.5(b). In the same way, a membership function  $\mu_{\text{about}}$  as shown in fig.6(a) quantifies the fuzziness "about".  $\mu_{\text{about}}$  evaluates the parallel deviation to the line segment  $\xi_1 \xi_2$  as shown in fig.6(b).

<sup>4</sup>Note that we can obtain these samples by the designer's maneuvering with remote controller.

<sup>5</sup> $x_{1,t} + x_{2,t} + \dots + x_{N,t} = 1$

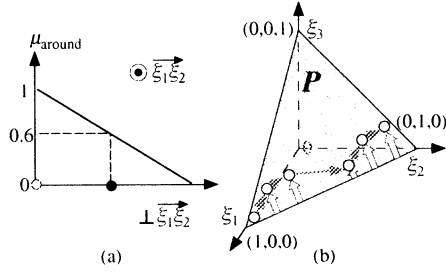


Fig.5  $\mu_{\text{around}}$

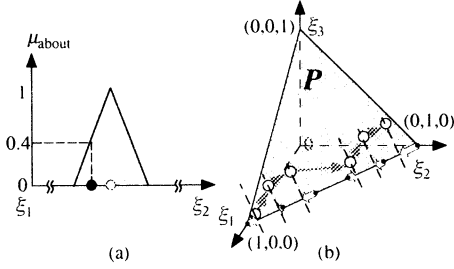


Fig.6  $\mu_{\text{about}}$

The fuzzification mentioned in subsec.2.1 gives the fuzzy environmental model  $\tilde{M}_S(\mathcal{S}, \mathcal{C}_S)$  and the membership functions defined in subsec.2.2 enables us to represent fuzzy sequential knowledge  $\tilde{s}_1^{t_f}$ . These ideas characterize the parameterization which defines a class of mathematical models for the continuous sensory/motor system.

### 3 Model Identification

#### 3.1 Problem Formulation

This subsection formulates the model identification. The information given here are sequential knowledge  $s_1^{t_f} = (\xi_1, \xi_2, \dots, \xi_{t_f})$  and  $L$  pairs of sampled inputs and outputs series  $u_1^{t_f, l} = (u_1^l, u_2^l, \dots, u_{t_f}^l)$  and  $y_1^{t_f, l} = (y_1^l, y_2^l, \dots, y_{t_f}^l)$ ,  $l = 1, 2, \dots, L$ , respectively<sup>6</sup>. As mentioned in the previous section, setting state equation  $F = (f_1, f_2, \dots, f_{t_f})$  and output equation  $G = (g_1, g_2, \dots, g_{t_f})$  defines the model. Given the input series  $u_1^{t_f}$ , the model calculates the state transition series  $x_1^{t_f} = (x_1, x_2, \dots, x_{t_f})$  and the output series  $y_1^{t_f} = (y_1, y_2, \dots, y_{t_f})$ . In this context, the model identification is formulated as the task to estimate parameters in  $F$  and  $G$  such that  $y_1^{t_f}$  would approximate  $y_1^{t_f, l}$  as good as possible on the condition

<sup>6</sup>To make the following discussion simple, we will consider a pair of inputs and outputs series and omit the notation of  $l$ .

that  $x_1^{t_f}$  follows the fuzzy constraint  $s_1^{t_f}$ <sup>8</sup>.

#### 3.2 Parameter Estimation Algorithm

To perform the parameter estimation formulated in the previous subsection, the following two performance indices:

$$E_f(x_1^{t_f}, F) = \sum_{i=1}^{t_f-1} \left| \Delta_i x_i - \sum_{i=1}^{\tau_f} x_{i,i} f_i(u_i) \right|^2 + h(x_1^{t_f}) \quad (6)$$

$$E_g(x_1^{t_f}, G) = \sum_{i=1}^{t_f} \left| y_i - \sum_{i=1}^{\tau_f} x_{i,i} g_i(u_i) \right|^2 + h(x_1^{t_f}) \quad (7)$$

are required to be minimized. The first term in eqn.(6) evaluates how the movement  $x_1^{t_f}$  should be on the convex polyhedron  $P$  and how  $F$  should be to realize the movement  $x_1^{t_f}$ . The second term denotes the fuzzy constraints<sup>9</sup> given by:

$$h(x_1^{t_f}) = \alpha \mu_{\text{around}}^{-1}(x_1^{t_f}) + \beta \mu_{\text{about}}^{-1}(x_1^{t_f}) \quad (8)$$

$$= \alpha \sum_{i=1}^{t_f} \left| x_i - x_i^\perp \right|^2 + \beta \sum_{i=2}^{t_f-1} \left| x_i^\perp - \frac{x_{i-1}^\perp + x_{i+1}^\perp}{2} \right|^2.$$

where  $x_i^\perp$  denotes the projection of  $x_i$  to the line segment  $\xi_1\xi_2$  (in the case of  $s_1: \xi_1 \rightarrow \xi_2$ ). The first term in eqn.(7) evaluates how the movement  $x_1^{t_f}$  should be on  $P$  and how  $G$  should be to output the appropriate  $y_1^{t_f}$ .

It requires to optimize  $x_1^{t_f}$  and the parameters in  $F$  and  $G$  simultaneously. The algorithm to realize the optimizations are derived from EM algorithm[6], where the optimization of  $x_1^{t_f}$  (1st step) and the optimization of  $F$  and  $G$  (2nd step) are alternately iterated. In the first step, on the assumption that  $F$  and  $G$  were already optimized as  $F^*$  and  $G^*$ ,  $x_1^{t_f}$  is updated in the direction to steepest gradient descent in  $E_f(x_1^{t_f}, F^*) + E_g(x_1^{t_f}, G^*)$ . In the second step, on the assumption that  $x_1^{t_f}$  were already optimized as  $x_1^{t_f*}$ , the parameters in  $F$  and  $G$  are updated in the direction to steepest gradient descent in  $E_f(x_1^{t_f*}, F)$  and  $E_g(x_1^{t_f*}, G)$ , respectively. This algorithm leads the set of parameters in  $F$  and  $G$  to one of the optima. It means that the local optimal model for the continuous sensory/motor system could be identified.

<sup>7</sup>Sequential knowledge  $s$  is implemented into the model as the form of fuzzy constraint which is quantified by membership functions  $\mu_{\text{around}}$  and  $\mu_{\text{about}}$ .

<sup>8</sup>Note that  $F$  and  $G$  must be differentiable with respect to every parameter of themselves.

<sup>9</sup>Note that the ranges of these membership grades are not  $[0, 1]$ . These are the reciprocal of the squared deviations defined in subsec.2.2. They are reasonable qualitatively.

## 4 Simulation and Discussion

Fig.7(a) shows the simulation environment, where two round obstacles are located and all sides are surrounded by walls. We supposed that the Markov environmental model is given by 6 discrete states denoted by  $\xi_1 \sim \xi_6$  as shown in fig.7(a), and the optimal plan is given as sequential knowledge  $s : \xi_1 \rightarrow \xi_2 \rightarrow \xi_4 \rightarrow \xi_3 \rightarrow \xi_5 \rightarrow \xi_6$ . Fig.7(b) shows the trajectories when the agent follows  $s$ . In the simulation, we specified coordinates of some representatives and got the basic route by spline interpolation, which is indicated by solid curve in fig.7(b). In fig.7(b),  $L = 10$  dotted curves were obtained also by spline interpolation with some representatives given from Gaussian distribution whose means are those of basic route. The agent has 5 distance sensors to detect the distance between the walls/obstacles and the agent itself, which are even allocated on the front of the agent from  $-90^\circ$  to  $90^\circ$  as illustrated at point A in fig.7(b). And it has the motor for rotating from  $-90^\circ$  to  $90^\circ$ . The agent navigates repeating sensation, rotation and moving a small distance.

We performed the identification of the model with parameters  $N = \tau_f = 6$ ,  $P = 5$ ,  $R = 1$ ,  $t_f = 100$ . In the simulation, hierarchical neural networks were used for  $f_i$  and  $g_i$ ,  $i = 1, 2, \dots, 6$ .

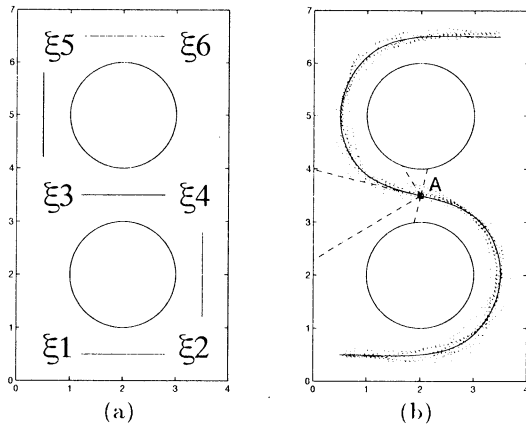


Fig.7 Environment

Fig.8(a) shows learning curves of  $E_f(x_1^{t_f}, F)$ ,  $E_g(x_1^{t_f}, G)$  and the sum of them <sup>10</sup>. They decreased more than 90 % by 10,000 times learning epochs. 100 curves in fig.8(b) represent the trajectories of 10-step <sup>11</sup> autonomous navigations from every 100 points on the basic route. From the result that most of the trajectories follows the basic route, we could say that the identified model almost succeeded in 10-step ahead

<sup>10</sup> Errors are normalized as those before learning are 1.0.

<sup>11</sup> 10-step is 10 % of the length of the basic route.

prediction, which could be highly evaluated as the model for dynamic systems.

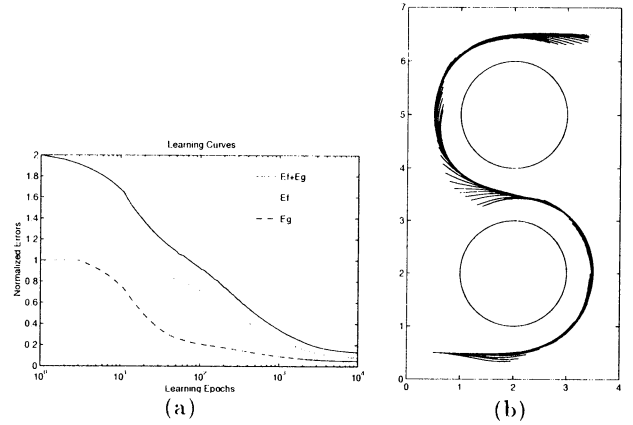


Fig.8 Result

## 5 Conclusion

This paper presented a model of the continuous sensory/motor systems of the agents. We extended the Markov environmental model and sequential plan to the discussion on topological space with implication from fuzzy modeling. Those extended ones characterized a class of mathematical models for the continuous sensory/motor system, which could be optimized by gradient descent-based algorithm. A simple simulation demonstrated the feasibility of the proposed method.

## References

- [1] S. Harnad. The symbol grounding problem. *Physica D*, Vol. 42, pp. 335–346, 1990.
- [2] M. Asada, S. Noda, S. Tawaratsumida, and K. Hosoda. Vision-based reinforcement learning for purposive behavior acquisition. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 146–153, 1995.
- [3] A. K. McCallum. *Reinforcement learning with selective perception and hidden state*. PhD thesis, Univ. of Rochester, 1996.
- [4] R. Kruse, R. Buck-Emden, and R. Cordes. Processor power considerations -an application of fuzzy markov chains. *Fuzzy Sets and Systems*, Vol. 21, No. 3, pp. 289–299, 1987.
- [5] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, Vol. 3, pp. 79–87, 1991.
- [6] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm (with discussion). *Journal of the Royal Statistical Society series B*, Vol. 39, pp. 1–38, 1977.



## Intelligent Fault Tolerant System of Vibration Control for Flexible Structures

Masahiro Isogai

Fumihito Arai, Toshio Fukuda

Yamazaki Mazak Co.  
Oguchi-cho, Aichi Pref., Japan

Nagoya Univ.  
Furo-cho, Chikusa-ku, Nagoya, Japan

### Abstract

This paper describes the intelligent fault tolerant control method for vibration control of flexible structures. When the fault phenomena of control system for the flexible structures can be supposed to be treated as the change of system parameters, the adaptive control based on neural network can be applied to the vibration control of flexible structures. By NN adaptive control system, input-output characteristics of plant and that of nominal model can be agreed, the decoupling control and linearizing control, which are based on the nominal model, can be used even if the change of system parameters are caused by fault. The effectiveness of the proposed fault tolerant control method is shown by the simulations of 5-link robotic arm.

### 1 introduction

As the scale of the structure become large, the fault rate of the control system arise, as the result of increasing in the number of the units such as sensors, actuators and controllers. When some failure occur, the performance of the vibration control goes down. It is necessary to consider the fault detection of units and reconfiguration of controllers.

In this study, the fault phenomena are supposed to be treated as parameter errors. In order to reconfigure the controller, the adaptive control based neural network is applied to the fault tolerant system for vibration control of flexible structures.

So far, various investigations have been carried out on the use of neural network for control system. Xu presented one of adaptive control using NN arranging NN parallel to the model with the parameter errors. This control system compensates the control inputs fed from the feedback controller designed for the nominal model, using the NN outputs. In this paper, this adaptive control system based on NN is applied to the flexible structures with change of parameters caused by faults. In this NN adaptive control system, the mod-

eling errors are compensated by the NN adaptive control. So the vibration can be controlled by the feedback loop of decoupling and linearizing control based on the model, which is arranged outside the NN adaptive control loop. The axial force is considered in the mathematical model treating as the outerforce to the vibration mode. By considering the axial forces in modeling, it is found that the mode dynamics is given as nonlinear system.

Finally, the effectiveness of the proposed control method is shown through simulations of 5-link robotic arm.

### 2 Mathematical modeling of flexible structures

In this paper, the flexible robotic arm is supposed to be an example of the flexible structures, of which base joint is fixed on the ground. The flexible links and the joints can move in the horizontal plane. Figure.1 illustrates the coordinate system of the flexible structures. The coordinate system of the first link is the absolute system, and that of the other links are the absolute system based in the balancing state of each link. The servomotors with reduction gears are installed at each joints in order to control the angle between links and the vibration of each link. The encoders for measurement of relative angles between links, and the tachogenerator for that of the relative angular velocity between links and the strain gauges for that of vibration are installed.

Vibration equations are given as following.

$$E_i I_i \frac{\partial^4 y_i}{\partial x_i^4} + \rho_i a_i \frac{\partial^2 y_i}{\partial t^2} + \frac{\partial}{\partial x_i} \{ S_i(x_i, t) \} \frac{\partial y_i}{\partial x_i} = 0 \quad (1)$$

where  $E_i I_i$ ,  $\rho_i$ ,  $a_i$ ,  $y_i$ ,  $S_i$  denote flexural rigidity, density, cross section, deflection, axial force of  $i$ th link, respectively.

Then using the mode expansion method, the deflec-

tion is expanded as follows.

$$y_i(x_i, t) = \sum_{j=0}^{\infty} \phi_{ij}(x_i) q_{ij}(t) \quad (2)$$

where  $\phi_{ij}(x_i)$ ,  $q_{ij}(t)$  are characteristic function and time function of  $i$ th link and  $j$ th mode, respectively.

Using boundary conditions, which are constructed by the rotational equations and the translational equations, the mode dynamics is given as follows.

$$\begin{aligned} \ddot{q}_{ij}(t) + \omega_{ij}^2 q_{ij}(t) &= \sum_{k=0}^{\infty} C_{i-1,k}^{i,j} q_{i-1,k} + \sum_{k=0}^{\infty} C_{i+1,k}^{i,j} q_{i+1,k} \\ &+ \sum_{k=0}^{\infty} F_{i,k}^a q_{i,k} \times \sum_{l=0}^{\infty} F_{i,l}^b q_{i,l} \\ &+ \sum_{k=0}^{\infty} F_{i,k}^c q_{i+1,k} \times \sum_{l=0}^{\infty} F_{i,l}^b q_{i,l} \\ &+ v_{ij} V_i + w_{ij} V_{i+1} \end{aligned} \quad (3)$$

Where  $C_{kh}^{ij}$  denotes the coupled force which is given to the  $i$ th mode of  $j$ th link from the  $k$ th mode of  $h$ th link. And it is found that 3rd and 4th terms in the right side of eq.(3), which are concerned with the axial force, are nonlinear.

### 3 Control method

#### 3.1 Control method for the nominal model

Eq.(3) can be written in the form of the state space method, as follows.

$$\dot{X} = AX + \eta(X) + BU \quad (4)$$

$$Y = CX \quad (5)$$

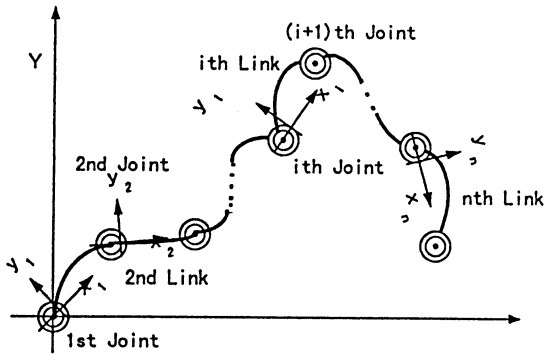


Fig.1 Coordinate system of flexible multi link system

Now separating of matrix  $A$  to matrix  $A_D$  and matrix  $A_C$

$$A = A_D + A_C \quad (6)$$

where matrix  $A_D$  means the decoupled matrix, and matrix  $A_C$  means the coupled matrix.

Here in order to treat this system as a linear and modally independent system, the following equations should be hold.

$$A_C - BK_c = 0 \quad (7)$$

$$\eta(X) - BK_l \eta(X) = 0 \quad (8)$$

So the gain  $K_c$  and the gain are given by

$$K_c = (B^T B)^{-1} B^T A_C \quad (9)$$

$$K_l = (B^T B)^{-1} B^T \quad (10)$$

The linearized and independent system is controlled based on the optimum control theory.

#### 3.2 NN adaptive control for the system included parameter errors

In this paper, NN adaptive control proposed by Xu is applied to vibration control for the flexible link system. As shown in Fig.2, NN is arranged parallel to the nominal model  $\hat{Y}(t)$  and the outputs of the identified model are given by the sum of the estimated output  $Y_{id}(t)$ , which is given through  $H_n(p)$ , and the model outputs  $Y_n(t)$ . NN is learned using  $e_{ID}(t)$  which is identification error between identified model outputs  $\hat{Y}(t)$  and the plant outputs  $Y_p(t)$ . The hierarchical type of NN is used as shown in Fig.3.

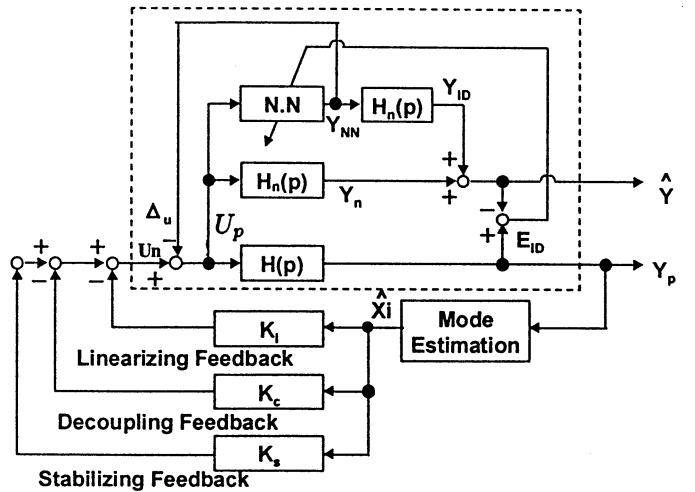


Fig.2 Block diagram of control system

The input vector  $U_{IN}^T(t)$  is

$$U_{IN}^T(t) = [ e_{m1}(t - \Delta t), e_{m2}(t - \Delta t), \dots, e_{mn}(t - \Delta t), \\ e_{m1}(t), e_{m2}(t), \dots, e_{mn}(t), \\ y_{p1}(t) \cdot y_{p2}(t), \\ y_{p2}(t) \cdot y_{p3}(t), \\ \dots, \\ y_{p(n-1)}(t) \cdot y_{pn}(t), \\ u_{p1}(t), u_{p2}(t), \dots, u_{pn}(t) ] \quad (11)$$

where

$e_{mi}(t)$ : errors between model outputs and plant outputs of  $i$ th link

$y_{p1}(t) \cdot y_{p2}(t)$ : the product of plant outputs of the neighboring links ; terms concerning the axial force

$u_{pn}(t)$ : the control inputs at the  $i$ th joint

And the relations between outputs and inputs of units are as follows.

$$y_{NN,k} = f(s_k) = 1/(1 + e^{-s_k}) \quad (12)$$

$$s_k = \sum_{j=1}^n v_{kj} \cdot h_j + \gamma_k \quad (13)$$

$$h_j = f(r_k) = 1/(1 + e^{-r_k}) \quad (14)$$

$$r_k = \sum_{j=1}^n w_{kj} \cdot I_j + \theta_k \quad (15)$$

where

$y_{NN,k}, h_j$ : the outputs of  $k$ th output layer unit, and that of  $j$ th unit of input layer

$s_k, r_k$ : the inner potential of the  $k$ th unit of the output layer, and that of  $k$ th unit of the hidden layer

$v_{kj}, w_{kj}$ : the coefficient of the connection between  $k$ th unit of the hidden layer and  $j$ th unit of the output layer, that of the connection between the output of the  $k$ th unit of the hidden layer and the output of the  $j$ th unit of the output layer

$\gamma_k, \theta_k$ : the threshold value of  $k$ th unit of the output layer, and that of  $k$ th unit of the hidden layer

$f(\cdot)$ : the sigmoid function

Using the identification errors  $e_{ID}$ , the performance function for NN to learn is defined as follows.

$$J(t) = \frac{1}{2} \sum_{k=1}^n e_{ID,k}(t)^2 \quad (16)$$

By backpropagation method [4], the weight coefficients of every layers are renewed so that this performance

function is minimized. The rules of renewing the weight coefficients are given concerning with the hidden layer and the input layer respectively as follows.

$$v_{jk}(t + \Delta t) = v_{jk}(t) \\ - \eta_1 \left\{ \sum_{i=1}^n (e_{ID,i} \frac{\delta}{\delta y_{NN,k}} y_{ID,i}) \right\} \\ \times \{ y_{NN,k} (1 - y_{NN,k}) h_j \} \quad (17)$$

$$w_{ji}(t + \Delta t) = w_{ji}(t) \\ - \eta_2 \left[ \sum_{k=1}^n \{ e_{ID,i} \frac{\delta}{\delta y_{NN,k}} (H_n(p) y_{NN,k}) \} \right. \\ \left. \times \{ y_{NN,k} (1 - y_{NN,k}) \} v_{jk} \right] h_j (1 - h_j) I_i \quad (18)$$

where  $\eta_1, \eta_2$  are the learning coefficients.

## 4 Simulations

Simulation of vibration control for the flexible 5 link robotic arm, which is an example of the flexible multi link structures, is carried out. The performance of the proposed control is simulated by commanding the angle of the joint, which is located in the end of the 5th link, from 0 degree to 30 degrees. In these simulations, the vibration mode of each link is considered for first mode.

To simulate the parameter errors caused by fault, it is supposed that the link length of the plant is 10% shorter than that of the nominal model.

Fig.5(a) shows the simulation result of control before NN learns, and Fig.5(b) shows that of control after NN learned.

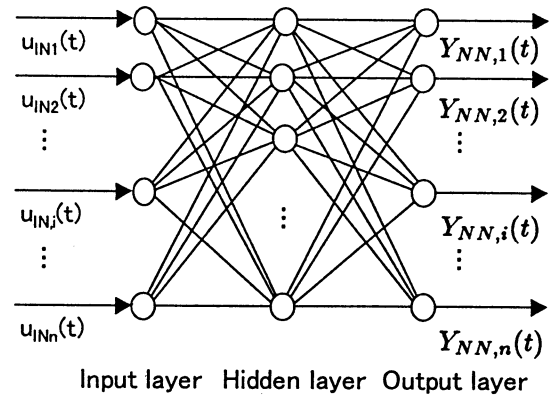


Fig.3 Hierarchical type of neural network

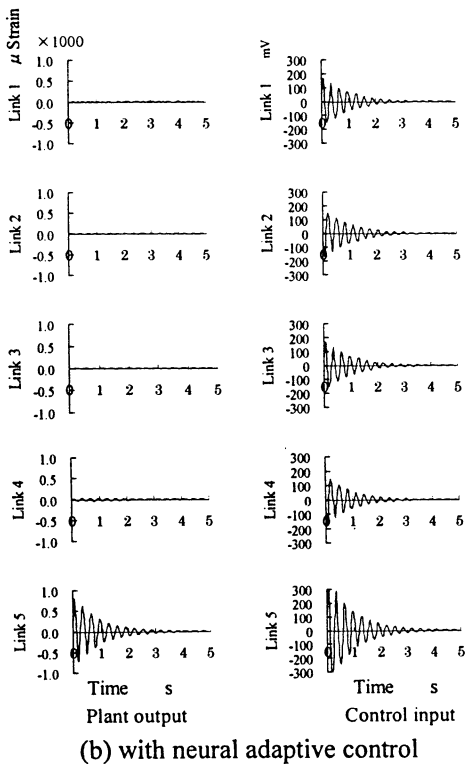
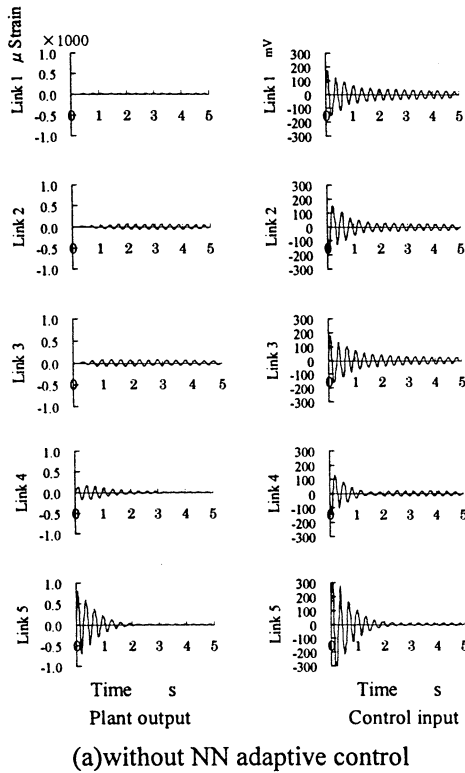


Fig.4 Simulation Results

By the result of control before NN is learning, it is found that vibration of 5<sup>th</sup> link propagates to other links, and the performance of decoupling control based on the nominal model goes down. On the other hand, by NN adaptive control, it is found that vibration of 5<sup>th</sup> link rarely propagates to other links. Since the modeling errors were compensated by the NN adaptive control, the linearizing and decoupling control based on the model could be performed effectively. By simulations, the effectiveness of the proposed method for the vibration control of the flexible structures is confirmed.

## 5 Conclusion

This paper proposed the fault tolerant control system for vibration control of flexible structures. Supposing that the fault phenomena is treated as the change of parameters, NN adaptive control was applied to control the vibration of the flexible structures with fault. Finally, the effectiveness of the proposed modeling and the control method was confirmed by simulations.

## References

- [1] T.Fukuda, M.Isogai(1987), Control of Coupling Vibrations in a Linked Flexible System(1st Report, Modeling of a linked flexible system and its basic control characteristics)(in Japanese). Trans. Jpn. Soc. Mech. Eng. Vol. 53, No.487, C, 664-670.
- [2] T.Fukuda, A.Arakawa(1987), Control of Flexible Robotic Arms(2nd Report, Modeling and Basic Control Characteristics of Second Degree-of-Freedom Coupling System)(in Japanese). Trans. Jpn. Soc. Mech. Eng. Vol. 53, No.488, C, 954-961.
- [3] B.B.Xu, T.Tsuji, M.Kaneko(1996), Torque Control of a Flexible Beam Using Neuro-Based Adaptive Control(in Japanese). Trans. Soc. Inst. Cont. Eng. 32, 4, 510-516.
- [4] Rumelhart, D.E., Hinton C.E. and Williams, R.J.(1986), Parallel Distributed Processing. The MIT Press, 1, 318-362.

# An Evolutionary Technique for Constrained Optimization Problems

M.M.A. Hashem\*, Keigo Watanabe\*\* and Kiyotaka Izumi\*\*\*

\*Faculty of Engineering Systems and Technology,  
Graduate School of Science and Engineering,

\*\*Department of Advanced Systems Control Engineering,  
Graduate School of Science and Engineering,

\*\*\*Department of Mechanical Engineering,  
Faculty of Science and Engineering,  
Saga University, 1-Honjomachi, Saga 840-8502, Japan

## Abstract

This paper deals with an evolutionary technique to solve the general Constrained Optimization Problems (COPs). This evolutionary technique is based on a Novel Evolution Strategy (NES) algorithm in which a Subpopulation-Based Arithmetical Crossover (SB-MAC) and a Time-Variant Mutation (TVM) operators were used. These operators came into view to be responded as specialized operators for handling constraints. A log-dynamic penalty function method is also used as a constraint violation measure for an individual. The effectiveness of this approach is demonstrated by solving some of the general test problems as well as by comparing with other evolutionary algorithms for these functions.

**KeyWords:** Evolutionary Algorithms; Constrained Optimization; Nonlinear Programming; Computational Intelligence

## 1 Introduction

The general constrained optimization problem COP for continuous variables  $\mathbf{x}$  is defined as [1]~[4]:

Minimize  $f(\mathbf{x})$  subject to the constraints

$$\begin{aligned} g_1(\mathbf{x}) &\leq 0, \dots, g_r(\mathbf{x}) \leq 0 \\ h_1(\mathbf{x}) &= 0, \dots, h_m(\mathbf{x}) = 0 \end{aligned} \quad (1)$$

where the *objective function*  $f$  and the  $g_i$ 's are functions on  $R^n$ , and the  $h_j$ 's are functions on  $R^n$  for  $m \leq n$ ,  $\mathbf{x} = [x_1, \dots, x_n]^T \in R^n$ , and  $\mathbf{x} \in \mathcal{F} \subseteq \mathcal{S}$ . A vector  $\mathbf{x}$  is called a *feasible* solution to COP if and only if  $\mathbf{x}$  satisfies the  $r + m$  constraints of COP. When the collection of feasible solutions is empty, COP is said to be *infeasible*. The set  $\mathcal{S} \subseteq R^n$  defines the search space and the set  $\mathcal{F} \subseteq \mathcal{S}$  defines a feasible part of the search space. Usually, the search space  $\mathcal{S}$  is defined as an  $n$ -dimensional rectangle in  $R^n$

$$x_l(i) \leq x_i \leq x_u(i), \quad 1 \leq i \leq n \quad (2)$$

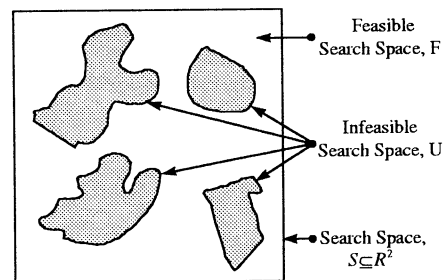


Figure 1: A typical two-dimensional search space  $\mathcal{S} \subseteq R^2$  and its feasible  $\mathcal{F}$  and infeasible  $\mathcal{U}$  parts

where  $x_l(i)$  and  $x_u(i)$  are lower and upper bounds for a variable  $x_i$  respectively, whereas the feasibility set  $\mathcal{F}$  is defined by an intersection of  $\mathcal{S}$  with constraint sets expressed by (1). In general, a search space  $\mathcal{S}$  consists of two disjoint subset of feasible and infeasible subspaces,  $\mathcal{F}$  and  $\mathcal{U}$ , respectively as shown in Fig. 1. At any point  $\mathbf{x} \in \mathcal{F}$ , the constraints  $g_i$  that satisfy  $g_i(\mathbf{x}) = 0$  are called the active constraints at  $\mathbf{x}$ . By extension, equality constraints  $h_j$  are also called active at all points of  $\mathcal{S}$ . However, it is worth mentioning that, without loss of generality, the maximization problems can be assumed only. If the optimization problem is to minimize a function  $f$ , this is equivalent to maximization a function  $z$ , where  $z = -f$ , i.e.,  $\min f(\mathbf{x}) = \max z(\mathbf{x}) = \max\{-f(\mathbf{x})\}$  [2].

Constrained optimization problems present the difficulties with potentially nonconvex or even disjoint feasible regions. Classic linear programming and nonlinear programming methods are often either unsuitable or impractical when applied to these constrained problems. Unfortunately, most of the real-world problems often pose such difficulties. Evolutionary algorithms are global methods, which aim at complex objective functions (e.g., non-differentiable or discontinuous) and they can be constructed to cope effectively with above difficulties. There are, however, no well-established guidelines on how to deal with infeasible

solutions.

Recently, several constraint-handling methods [1]~[4] for continuous numerical optimization problems by evolutionary computation techniques have emerged in case of  $\mathcal{F} \subseteq \mathcal{S}$  (Fig. 1). These can be classified into several categories, such as methods based on 1) penalty functions, 2) preserving feasibility of solutions, 3) a search for feasible solutions, and 4) hybrid methods. Among these methods, most popular one is penalty function method in which sometimes static, dynamic, annealing, adaptive or death penalties are used. Each of these methods, of course, has its own merits and demerits for nonlinear programming problems.

In this paper, a Novel Evolution Strategy (NES) algorithm [6], which was developed in the framework of ESs [2, 5], is used as a basic evolutionary technique for the constrained optimization problems. In this algorithm, a Subpopulation-Based Arithmetical Crossover (SBMAC) and a Time-Variant Mutation (TVM) operators were used. These operators came into view to be responded as specialized operators for handling constraints. A log-dynamic penalty function method is also used as a constraint violation measure for an individual. The effectiveness of this approach is demonstrated by solving some of the test problems [1]~[4] as well as by comparing with other evolutionary algorithms for these functions.

## 2 Specialized Variation Operators

It seems that one of the most reasonable heuristics for dealing with the issue of feasibility is to use specialized variation (genetic) operators [2] to maintain the feasibility of individuals in the population. Here, such approaches are discussed which was used in the NES algorithm.

### 2.1 Subpopulation-Based Max-mean Arithmetical Crossover (SBMAC)

A Subpopulation-Based Max-mean Arithmetical Crossover (SBMAC) [6] was employed in the crossover in which each subpopulation's elite and the mean-individual created from that subpopulation excluding the elite were used. It has a very strong directivity to the elite as shown in **Fig. 2**. The SBMAC explores promising areas in the  $\mathcal{S}$  with different directivity towards the optimum point while avoiding the constraints. And there is less possibility of being trapped in local minima as well as of violating the constraints while attempting to attain the optimum.

The population consisting of  $\mu$  individuals is divided into  $l$  subpopulations in each generation such that each subpopulation will have  $\frac{\mu}{l}$  individuals. The  $\psi_{j,\max}$  is defined as an elite individual that maximized a cost function within the  $j$ -th subpopulation, and a mean-individual  $\bar{\psi}_j$  is created from the  $j$ -th subpopulation

excluding the  $\psi_{j,\max}$ . Now the crossover to produce two offspring ( $\zeta_1, \zeta_2$ ) is defined as

$$\zeta_1 = \alpha\psi_{j,\max} + (1-\alpha)\bar{\psi}_j \quad (3)$$

$$\zeta_2 = (1-\alpha)\psi_{j,\max} + \alpha\bar{\psi}_j \quad (4)$$

where  $\alpha$  is selected from URN[0,1]. Note that  $\alpha$  is sampled anew for each object variable of the individuals and for the  $j$ -th subpopulation  $\frac{\mu}{l}$  offspring are generated. Thus,  $\mu$  number of offspring are generated for the  $l$  subpopulations in each generation.

It is, however, worth noting that if the individuals  $\psi_{j,\max}$  and  $\bar{\psi}_j$  are feasible, then they will yield always feasible individuals for  $0 \leq \alpha \leq 1$  in the convex  $\mathcal{S}$ . Consequently, the dependency on the penalty function will be relaxed and the search process will be faster as well.

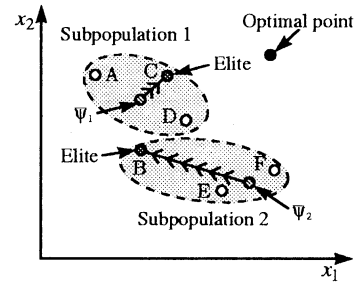


Figure 2: An example of the subpopulation-based max-mean arithmetical crossover

### 2.2 Time-Variant Mutation (TVM)

This special dynamic TVM [6] operator is designed aiming at both improving the fine local tuning and reducing the disadvantage of uniform mutation [2, 5]. The TVM is defined for a child  $\zeta_j^t$  as

$$x'_i = x_i + \sigma(t) \cdot N_i(0, 1) \quad \forall i \in \{1, \dots, n\} \quad (5)$$

where  $N_i(\cdot, \cdot)$  indicates that the Gaussian random value with zero-mean and unity variance and it is sampled anew for each value of the index  $i$ , and  $\sigma(t)$  is the time-variant mutation step at the generation  $t$ , which is defined by

$$\sigma(t) = \left[ 1 - r^{(1 - \frac{t}{T})^\gamma} \right] \quad (6)$$

where  $r$  is selected from URN[0,1],  $T$  is the maximal generation number,  $\gamma$  is a parameter determining the degree of dependency on the generations. The function  $\sigma(t)$  returns a value in the range [0,1] such that the probability of  $\sigma(t)$  being close to 0 increases as age of the population,  $t$  increases. This property of  $\sigma(t)$  causes to search the problem space uniformly initially (when  $t$  is small) and very locally at  $t$  larger stages.

## 3 Constrained Fitness Function

In Evolutionary Computation (EC) as in many other fields of optimization, most of the constraint-handling methods are based on the concept of penalty

functions [1]~[4], which usually penalize infeasible individuals. They differ, however, in how the penalty function is designed and applied to infeasible solutions. The cost function  $f$  is usually used to evaluate a feasible solution, i.e.,

$$\Phi_f(\mathbf{x}) = f(\mathbf{x}), \quad \text{for } \mathbf{x} \in \mathcal{F} \quad (7)$$

and the constraint violation measure  $\Phi_u(\mathbf{x})$  for  $r + m$  constraints, usually defined as [1, 2]

$$\Phi_u(\mathbf{x}) = \left[ \sum_{i=1}^r (g_i^+(\mathbf{x}))^2 + \sum_{j=1}^m (h_j(\mathbf{x}))^2 \right] \quad (8)$$

where  $g_i^+(\mathbf{x})$  and  $h_j(\mathbf{x})$  are the magnitude of the violation of inequality and equality constraints respectively. Here,  $g_i^+(\mathbf{x})$  is defined as  $\max\{0, g_i(\mathbf{x})\}$ , where  $1 \leq i \leq r$ . Then the total evaluation of an individual, which can be interpreted as the error (for a minimization problem) or fitness (for a maximization problem) of an individual to COP is obtained as

$$\Phi(\mathbf{x}) = \Phi_f(\mathbf{x}) + \frac{s_t}{2} \Phi_u(\mathbf{x}) \quad (9)$$

where  $s_t$  is a penalty parameter given by an iteration.

Now the complete evaluation function for an individual becomes as

$$\Phi(\mathbf{x}) = f(\mathbf{x}) + \frac{s_t}{2} \left[ \sum_{i=1}^r (g_i^+(\mathbf{x}))^2 + \sum_{j=1}^m (h_j(\mathbf{x}))^2 \right] \quad (10)$$

A primary concern for the penalty function approach is how to assign an appropriate penalty parameter  $s_t$  to the constraints. The penalty function theorem [1] provides a guideline to the above method on how the penalty parameter  $s_t$  should be selected. This theorem implies that the penalty parameter  $s_t$  should be of increasing nature other than constant. Thus a log-dynamic penalty parameter  $s_t$  is defined as

$$s_t = s_0 + \log(t + 1) \quad (11)$$

where  $s_0$  is the initial value of the penalty parameter and  $t$  is the current generation number.

## 4 The Evolutionary Algorithm

The Novel Evolution Strategy (NES) [6] which utilizes the above described specialized variation operators and constrained fitness function is shown in **Fig. 3** and discussed below briefly:

The initial population consisting of  $\mu$  individuals is generated using a URN within desired bounds of the object variables  $\mathbf{x}$ . After evaluating the  $\mu$  individuals with the fitness function  $\Phi(\mathbf{x})$  and ordering them according to their function values, this population is considered as parents for the next generation. These parents undergo evolutionary operations described above to create  $\mu$  children. These children population are used to evaluate the fitness function  $\Phi(\mathbf{x})$  and they are then added to the parent population, and the best  $\mu$  individuals from the combined parent and children population are selected as parents to form the next generation. This process is continued until the maximum generation  $T$  is exhausted.

```

Algorithm NES()
{
    t=0; /* Initialize the generation counter */
    Initialize_Population();
    Evaluate_Population();
    while(t ≠ T) do{
        Apply_SBMAC(); /* Crossover operation */
        Apply_TVM(); /* Mutation operation */
        Evaluate_Population();
        Alternate_Generation();
        t++; /* Increase the generation counter */
    }
}

```

Figure 3: The structure of the NES algorithm

## 5 Test Problems

The developed evolutionary algorithm is tested against the following test problems which include quadratic and nonlinear functions with several linear and nonlinear constraints. These problems were used as the test cases for the Two Phase EP (TPEP) [1] and for the GENOCOP II system [2]~[4].

**Problem 1:** Minimize  $f(\mathbf{x}) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$  subject to nonlinear constraints

$$\begin{aligned} c_1 &: x_1 + x_2^2 \geq 0 \\ c_2 &: x_1^2 + x_2 \geq 0 \end{aligned}$$

and bounds  $-0.5 \leq x_1 \leq 0.5$  and  $x_2 \leq 1.0$ . The known global solution is  $\mathbf{x}^* = (0.5, 0.25)$  and  $f(\mathbf{x}^*) = 0.25$ .

**Problem 2:** Minimize  $f(\mathbf{x}) = -x_1 - x_2$  subject to nonlinear constraints

$$\begin{aligned} c_1 &: x_2 \leq 2x_1^4 - 8x_1^3 + 8x_1^2 + 2 \\ c_2 &: x_2 \leq 4x_1^4 - 32x_1^3 + 88x_1^2 - 96x_1 + 36 \end{aligned}$$

and bounds  $0 \leq x_1 \leq 3$  and  $0 \leq x_2 \leq 4$ . The known global solution is  $\mathbf{x}^* = (2.3295, 3.1783)$  and  $f(\mathbf{x}^*) = -5.5079$ .

**Problem 3:** Minimize  $f(\mathbf{x}) = (x_1 - 10)^3 + (x_2 - 20)^3$  subject to nonlinear constraints

$$\begin{aligned} c_1 &: (x_1 - 5)^2 + (x_2 - 5)^2 - 100 \geq 0 \\ c_2 &: -(x_1 - 6)^2 - (x_2 - 5)^2 + 82.81 \geq 0 \end{aligned}$$

and bounds  $13 \leq x_1 \leq 100$  and  $0 \leq x_2 \leq 100$ . The known global solution is  $\mathbf{x}^* = (14.095, 0.84296)$  and  $f(\mathbf{x}^*) = -6961.81381$ .

**Problem 4:** Minimize  $f(\mathbf{x}) = 0.01x_1^2 + x_2^2$  subject to nonlinear constraints

$$\begin{aligned} c_1 &: x_1x_2 - 25 \geq 0 \\ c_2 &: x_1^2 + x_2^2 - 25 \geq 0 \end{aligned}$$

and bounds  $2 \leq x_1 \leq 50$  and  $0 \leq x_2 \leq 50$ . The global solution is  $\mathbf{x}^* = (\sqrt{250}, \sqrt{2.5}) = (15.811388, 1.581139)$  and  $f(\mathbf{x}^*) = 5.0$ .

**Problem 5:** Minimize  $f(\mathbf{x}) = (x_1 - 2)^2 + (x_2 - 1)^2$  subject to a nonlinear constraint

$$c_1 : -x_1^2 + x_2 \geq 0$$

and a linear constraint

$$x_1 + x_2 \leq 2$$

The global solution is  $\mathbf{x}^* = (1.0, 1.0)$  and  $f(\mathbf{x}^*) = 1.0$ .

Table 1: A performance comparison of the evolutionary technique with TPEP and GENOCOP II systems

Problem No.	GENOCOP II System	TPEP System	NES Algorithm			
			$b$	$m$	$w$	$\sigma$
1	0.25000	0.25000	0.25000	0.25000	0.25000	$6.00555 \times 10^{-7}$
2	-5.50790	-5.50801	-5.50801	-5.50801	-5.50799	$2.77307 \times 10^{-6}$
3	-6961.81381	-6961.81388	-7950.96189	-7950.96189	-7950.96189	$1.59349 \times 10^{-7}$
4	5.00000	5.00000	5.00000	5.00014	5.00264	$7.52144 \times 10^{-4}$
5	1.00000	1.00000	1.00000	1.00000	1.00002	$2.90458 \times 10^{-6}$

## 6 Implementation, Results and Discussions

The algorithm was implemented using  $\mu = 60$ ,  $l = 10$  and  $\gamma = 8.0$  for all the problems. The maximal generation number  $T$  was selected as 100, 200, 700, 1000 and 90, and the initial penalty parameter  $s_0$  was selected as  $1 \times 10^3, 1 \times 10^7, 1 \times 10^9, 1 \times 10^5$  and  $1 \times 10^5$  for the problems 1, 2, 3, 4 and 5 respectively. For each problem, the constraints were converted to the form of (1) before solving the problem. The algorithm has been executed on the test problems 30 times with different sample paths. The obtained results are summarized in Table 1 where the best ( $b$ ), median ( $m$ ), worst ( $w$ ) and the solution standard deviation ( $\sigma$ ) results are reported and compared with the TPEP and GENOCOP II systems produced results.

Among all the solutions, the evolution history produced by the algorithm for the Problem 1 is shown in Fig. 4, and it indicates the perfect constraint satisfaction at the closing end of the evolution, i.e.,  $\Phi(\mathbf{x}) = f(\mathbf{x})$ . The algorithm produced similar results as that of TPEP and GENPCOP II systems for problems 1, 4 and 5. The algorithm produced similar to TPEP result but better than GENOCOP II system for problem 2. The solution is  $\mathbf{x}^* = (2.32952020, 3.17849313)$ , and  $f(\mathbf{x}^*) = -5.5080133$ , which is lower than GENOCOP II result and feasible. For problem 3, it provided with a better solution than previous known solution as well as the solution obtained by TPEP and GENOCOP II systems. The solution found for problem 3 is:  $\mathbf{x}^* = (1.36602541e + 01, 4.76275753e - 16)$ , at which  $c_1 = 1.076567e - 06$  and  $c_2 = 2.181000e + 01$ , and  $f(\mathbf{x}^*) = -7950.961892$ , which is much more lower than the previous known solution ( $-6961.81388$ ) and feasible. The lower  $\sigma$  value indicate that the algorithm responded with a high convergence reliability. The convergence velocity of the algorithm is faster than that of GENOCOP II system which consumed 1000 generations for all the problems.

## 7 Conclusions

The proposed algorithm is seemed to be simple as compared to TPEP and GENOCOP II systems. The global results with high precision were achieved by virtue of its fine local tuning capability. Like other

evolutionary algorithms, it did not require any implicit/explicit calculations of gradient or Hessian matrix of the objective function and constraints. Consequently, proposed method did not suffer from the ill-conditioned Hessian problem usually associated with some calculus-based methods. The performance of the algorithm seemed to be an efficient method for constrained optimization problems.

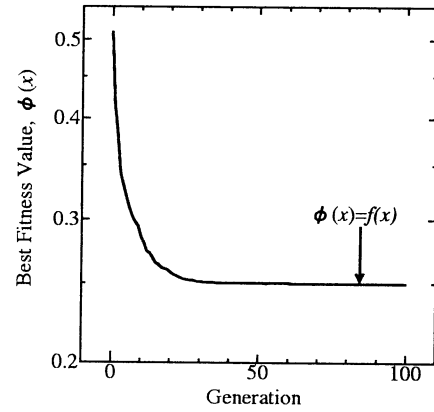


Figure 4: An evolution history of the algorithm for the Problem 1

## References

- [1] Kim, J.-H. and Myung, H., "Evolutionary Programming Techniques for Constrained Optimization Problems," IEEE Trans. on Evolutionary Computation, Vol. 1, No. 2, pp. 129-140, (1997).
- [2] Michalewicz, Z., "Genetic Algorithms + Data Structures = Evolution Programs," 3rd. rev., and extended edition, Springer-Verlag, Berlin, (1996).
- [3] Michalewicz, Z., "Evolutionary Computation Techniques for Nonlinear Programming Problems," International Trans. on Operation Research, Vol. 1, No. 2, pp. 223-240 (1994).
- [4] Michalewicz, Z. and Attia, N. F., "Evolutionary Optimization of Constrained Problems," Procs. of the 3rd. Annual Conf. on Evolutionary Programming, River Edge, NJ, World Scientific, pp. 98-108 (1994).
- [5] Bäck, T. and Schwefel, H.-P., "An Overview of Evolutionary Algorithms for Parameter Optimization," Evolutionary Computation, Vol. 1, No. 1, pp. 1-23, (1993).
- [6] Hashem, M.M.A., Watanabe, K. and Izumi, K., "A New Evolution Strategy and Its Application to Solving Optimal Control Problems," JSME International Journal, Series C, Vol. 41, No. 3, pp. 406-412, (1998).



# Evolving in Dynamic Environments Through Adaptive Chaotic Mutation

D.P.Thrishantha Nanayakkara†

Keigo Watanabe†

Kyotake Izumi‡

† Dpt. of Graduate School of Science and Engineering, Saga University, Saga 840-8502

‡Dpt. of Mechanical Engineering, Saga University, Saga 840-8502.

## Abstract

A new approach to incorporate the features and robustness of the natural evolutionary process to evolutionary computation is proposed. The key feature of this approach is maintaining the evolvability of the population by controlling the mutation process by exploiting the characteristic dynamics of a chaotic neuron. The proposed method has been tested for optimizing in static environments as well as in a changing environment. The results show that it inherits the ability to cope with the change in the optimizing criteria and converges fast while maintaining evolvability.

**Keywords:** Chaotic Neuron, Evolutionary Programming, Dynamic Environments.

## 1 introduction

Evolutionary computation has been given much attention in the recent history of engineering optimization problems. Though evolutionary computation is effective in finding a globally optimum solution to a static optimization criteria, work is yet to be done to enhance its ability to cope with a dynamic environment, Bäck [1]. In fact, actual engineering problems, are sometimes coupled with phenomena such as emergence and disappearance of constraints, changing optimization criteria, *etc.* Therefore evolutionary computation needs to take a turn towards incorporating the feature of evolvability in dynamic environments.

This paper proposes a new concept that takes a step towards natural evolution. It is important to be noted that the ability of a population to continue surviving in a changing environment comes from the ability to find the global optimum in the current environment, while maintaining enough diversity to face a potential change in the environment. In the proposed method, this phenomenon is simulated by making use of the characteristics of a chaotic neuron proposed by Aihara *et al.* [2].

The new approach of chaotic mutation gives promising results for some benchmark functions. In the sections to come, the meta-evolutionary programming algorithm is taken as the conventional method of evolutionary computation. All comparisons are being made with this algorithm.

## 2 Meta-EP Algorithm

### 2.1 The Representation

In evolutionary programming (EP), the individuals are defined based on real valued vectors in the form

$$\mathbf{a}_j = (\mathbf{x}_j, \boldsymbol{\sigma}_j), \quad j = (1 \dots \mu)$$

to deal with continuous parameter optimization problems. Here  $\mathbf{x}$  is the object variable vector and the strategy parameters  $\boldsymbol{\sigma}_j$  are used for the mutation of individuals as explained later.

### 2.2 Recombination

Recombination is not found in meta-EP algorithm. In the proposed method arithmetical crossover is adopted with a relatively small probability (0.2). It is basically performed as

$$x'_{S,i} = x_{S,i} + \chi(x_{S,i} - x_{L,i})$$

$$x'_{L,i} = x_{L,i} + \chi(x_{L,i} - x_{S,i})$$

where  $S$  and  $L$  denote two parent individuals selected at random from the parent population, and  $\chi \in [0, 1]$  is a uniform random variable.

### 2.3 Mechanism of Mutation

Mutation is performed to each individual in the form,

$$x'_i = x_i + \sigma_i N_i(0, 1) \quad (1)$$

where  $x_i$  is the object variable of a randomly selected individual.  $\sigma_i$  is the standard deviation or so called strategy parameter, and  $N_i(0, 1)$  is a Gaussian random value generator with a normal distribution of zero mean and unity variance.

In the case of meta-EP, the standard deviation is controlled as

$$x'_i = x_i + \sqrt{\nu_i} N_i(0, 1). \quad (2)$$

The variance  $\nu_i$  is a constant and modified as follows.

$$\nu'_i = \nu_i + \sqrt{\nu_i \alpha} N_i(0, 1) \quad (3)$$

where the value of  $\alpha$  is an exogenous parameter that makes sure that the variance remains positive. In the case where the variance becomes negative or zero, it is set to a small value  $\epsilon > 0$ . As can be seen in equations (1), (2) and (3), the drawback in evolutionary programming is that the standard deviation that is gradually being changed is blind to the diversity of the population. As such the method is good in dealing with static environments but lacks the robustness to work in dynamic environments.

### 2.4 Selection Mechanism

In meta-EP, after creating  $\mu$  offspring from  $\mu$  parent individuals by mutating each parent once, tournament selection mechanism is adopted to produce the next generation. In the tournament selection method for each individual  $\mathbf{a}_j \in P(t) \cup P'(t)$ , where  $P(t)$  is the parent population and  $P'(t)$  is the population of mutated individuals, a random uniform sample of size  $T$  is picked up from the population such that  $T > 1$ . Then a score  $w_j \in (0, T)$  is attributed to each  $\mathbf{a}_j$  such that the score is equal to the number of individuals in  $T$  that is less fitter than the individual. Based on this score, the  $2\mu$  individuals are ranked and the best  $\mu$  individuals are selected for the next generation.

## 3 Adaptive Chaotic Mutation Mechanism

The standard deviation in the proposed mutation mechanism is controlled depending on the instantaneous diversity of the population by exploiting the characteristics of a chaotic neuron proposed by Aihara *et al.* [2], and adopted appropriately for the purpose. An extended

version of this chaotic neuron is adopted by Choi *et al.* [3], to avoid local minimal problem in mobile robots. The fundamental idea of avoiding local minimums is almost the same in this method.

Before going into a detailed discussion let us briefly look at the characteristics of the chaotic neuron model used here.

### 3.1 Characteristics of a Chaotic Neuron

The following equations are used to elaborate the characteristics of the Aihara's chaotic neuron.

$$p(n+1) = f(q(n+1)) \quad (4)$$

$$q(n+1) = kq(n) - \beta g(p(n)) + u(n) \quad (5)$$

$$f(q) = \frac{\phi}{1 + e^{-q/\psi}} - \frac{\phi}{2} \quad (6)$$

where  $p(n+1)$  is the output value of the neuron at discrete time  $n+1$ , which lies between  $\phi/2$  and  $-\phi/2$ ,  $\psi$  controls the shape of the function  $f(q)$  while  $\phi$  determines the magnitude of it. In addition  $\beta$  is a positive parameter,  $k$  is the damping factor of refractoriness.  $u(n)$  is the input to the chaotic neuron.

The refractory function  $g()$  is given by

$$g(p(n)) = \begin{cases} \frac{1}{2\pi\rho^2} e^{-\|p(n)\|^2/2\rho^2} & \text{if } p(n) - p(n-1) \geq 0 \\ -\frac{1}{2\pi\rho^2} e^{-\|p(n)\|^2/2\rho^2} & \text{Otherwise.} \end{cases} \quad (7)$$

The value of  $\rho$  controls the shape of  $g(p(n))$ . It can be seen in Fig. 1 that the output undergoes chaotic motions when the input  $u(n)$  draws closer to zero. The reason lies in the shape of the refractory function which is sensitive to the sign of the gradient of output.

### 3.2 Application to Adaptively Mutate Individuals

The relative diversity is defined as

$$\zeta_n = \frac{D(n)}{D(0)} \quad (8)$$

where  $D(n)$  is the population diversity [4] of the  $n^{\text{th}}$  generation, which is given by

$$D(n) = \sqrt{\frac{\sum_{j=1}^N \{fit_m(n) - fit_j(n)\}^2}{N}} \quad (9)$$

In which  $fit_m(n)$  is the mean fitness of the  $n^{\text{th}}$  generation,  $fit_j(n)$  is the fitness of the  $j^{\text{th}}$  individual in the

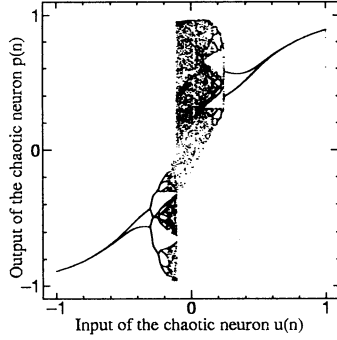


Figure 1: Behavior of the output of the chaotic neuron with the input  $u(n)$  used as the bifurcation parameter. The parameter values adopted in this application are,  $k=0.3$ ,  $\beta=0.5$ ,  $\phi=2.0$ ,  $\psi=0.5$ ,  $\rho=0.2$ .

$n^{\text{th}}$  generation and  $N$  denotes the number of individuals. Then

$$a(n) \equiv \zeta_n$$

The relative diversity decreases gradually as the population converges to a global optimum and it gradually decreases the standard deviation due to the characteristics of the chaotic neuron. When the relative diversity is near zero, the standard deviation undergoes chaotic motions regaining diversity and enhancing further evolvability. Therefore the modified version of mutation mechanism proposed in this approach is

$$x'_i = x_i + \sigma_i(\zeta_n)N_i(0, 1) \quad (10)$$

where the chaotic standard deviation  $\sigma_i(\zeta_n)$  is defined by

$$\sigma_i(\zeta_n) = \theta_i \{ (1 + U([0, 1])) / 2 \} \vartheta(\zeta_n) \quad (11)$$

$$\theta_i = cB_i \quad (12)$$

in which the function of chaotic neuron is given by  $\vartheta$ ,  $B_i$  is the range of initialization of the object variable  $x_i$ ,  $c$  is set to be 0.125. The value of  $c$  can be altered depending on the degree of dynamics in the environment.  $U([0, 1])$  is the uniform Gaussian random number generator between  $[0, 1)$ .

#### 4 Optimization of a Moving Function

For the testing of the proposed method in its ability to optimize in a dynamic environment, the following moving

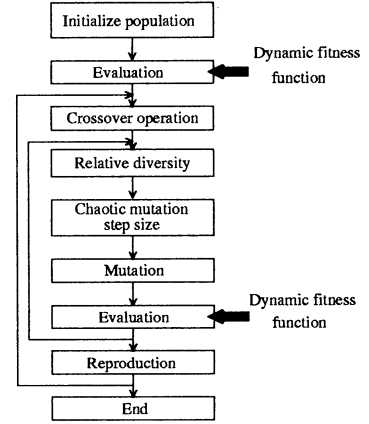


Figure 2: The block diagram showing the proposed adaptive chaotic mutation method

function was considered.

$$H(n, x, y) = \begin{cases} 1 + xe^{(-x^2 - y^2)} & \text{if } 0 < n \leq 100 \\ 1 + (x + 0.2)e^{(-(x+0.2)^2 - y^2)} & \text{if } 100 < n \leq 200 \\ 1 + (x + 0.3)e^{(-(x+0.3)^2 - y^2)} & \text{if } 200 < n \leq 300 \\ 1 + (x + 0.4)e^{(-(x+0.4)^2 - y^2)} & \text{if } 300 < n \leq 400 \\ 1 + (x + 0.5)e^{(-(x+0.5)^2 - y^2)} & \text{if } 400 < n \leq 450 \end{cases} \quad (13)$$

where  $n$  is the number of generations. In this dynamic environment, the test results of the proposed method for tracking the optimum point are shown. A population size of 60, crossover probability of 0.2, tournament size of 10 were used in the proposed method. In the conventional method, same parameters except crossover were used. The population was initialized in  $(-2, 2)$ . It can be seen

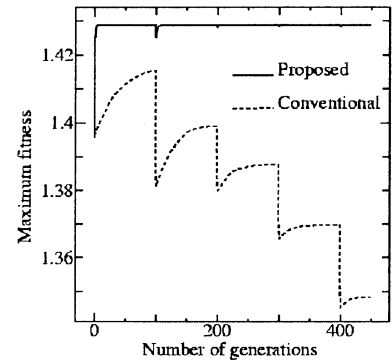


Figure 3: The behavior of the maximum fitness in the proposed method and evolutionary programming method, averaged over 100 trials.

that in the conventional method the ability to track the optimum solution deteriorates when the function moves, compared with the proposed method.

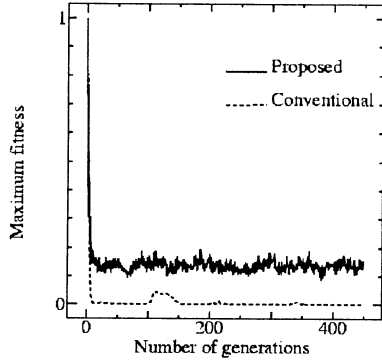


Figure 4: The behavior of the relative population diversity in the proposed method and evolutionary programming method, averaged over 100 trials.

## 5 Optimizing static environments

Two bench mark functions have been used to demonstrate the convergence ability and the reliability of finding the global optimum in a static environment using the proposed method. A population size of 50, crossover probability of 0.2, tournament size of 10 was used in the proposed method. In the conventional method, same parameters except crossover was used. For the function  $f_1$  the population was initialized in  $(-30, 30)$  and for  $f_2$  in the range  $(-1.28, 1.28)$

$$f_1(\mathbf{x}) = \sum_{i=1}^{30} x_i^2 \quad (14)$$

$$f_2(\mathbf{x}) = \sum_{i=1}^{30} x_i^4 + U([0, 1)) \quad (15)$$

As can be seen in Figs. 5 and 6, in both cases, the proposed method has shown superior performance than the conventional meta-EP method.

## 6 Summary

A new concept of adaptive mutation with the objective of optimization in dynamic environments has been proposed. The new approach differs from the other methods in considering the diversity of the population as the control input to a chaotic neuron to control the standard deviation of mutation. The process works in observing the diversity and then uses the relative diversity as the input to the chaotic neuron and decides on a suitable standard deviation for mutation.

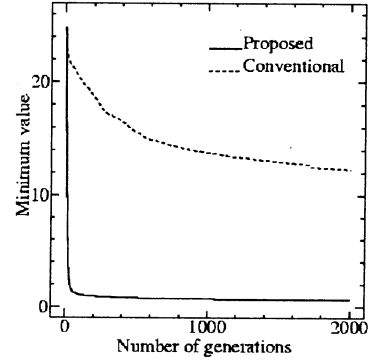


Figure 5: Comparison of the behavior of the best fit value averaged over 100 trials for the quadratic function  $f_1(x)$  between the proposed and conventional methods.

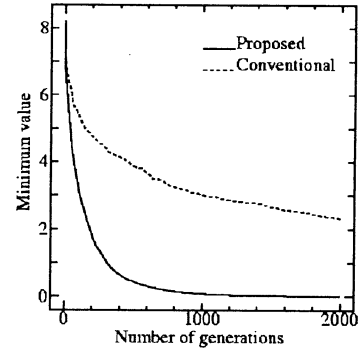


Figure 6: Comparison of the behavior of the best fit value averaged over 100 trials for the noisy quadratic function  $f_2(x)$  between the proposed and conventional methods.

## References

- [1] T. Bäck and H. P. Schwefel, "An Overview of Evolutionary Algorithms for Parameter Optimization," *Evolutionary Computation*, vol. 1, No. 1, pp. 1-23, 1993.
- [2] K. Aihara, T. Takabe and M. Toyoda, "Chaotic Neural Networks," *Physics letters, A*, vol. 144, nos. 6,7, pp. 333-340, 1990.
- [3] Changkyu Choi and Ju-Jang Lee, "Dynamical Path-Planning Algorithm of a Mobile Robot: Local Minima Problem and Nonstationary Environments," *Mechatronics*, Vol 6, No. 1, pp. 81-100, 1996.
- [4] M.M.A. Hashem, K. Watanabe and K. Izumi, "A Stable-Optimum Gain Tuning Method for Designing Mobile robot Controllers Using an Incest Prevented Evolution Strategy," *Jouranal of Advanced Computational Intelligence*, to be published, 1999.

## Organizational Evolution by Learning Adaptive Functions

Masayuki Ishinishi & Akira Namatame

Dept. of Computer Science

National Defense Academy

Yokosuka, 239-8686, JAPAN

E-mail: mishi@cs.nda.ac.jp, nama@cc.nda.ac.jp

### Abstract

In this paper, we aim at providing a general theoretical framework for designing complex adaptive systems as a society of rational agents. We term entities with their own interest as agents. They are also rational in the sense that they only do what they want to do and what they think is in their own best interest. We formulate the dynamic interaction among those rational agents as the competitive and cooperative problems. We obtain the equilibrium behavior in the long-run, and characterize collective behavior of those rational agents. We show that how the complex collective behavior can be emerged from locally optimal behavior of each agent. We also describe why and how they organize themselves into the multi-level hierarchical organization with the nesting structures.

Key words:

emergent collective behaviors, market mechanism, economic agent, emotion

## 1 Introduction

In a large-scale complex adaptive system composed of those many rational agents, two types of strategic behaviors may occur: agents mutually interact and behave as a group to optimize the goal of a group, while at the same time, each rational agent also behaves to optimize its own adaptive or object. For an individual rational agent, it behaves to improve its own adaptive based on its local observation. This ability is based on principles of the individual rationality. By a social goal we mean a goal that is not achievable by any single agent alone but is achievable by a group of agents.

We describe the economic-oriented interactions among agents as the basis for social cooperation learnable through local interaction. We call the latter ability as competitive cooperation. We ask the following questions about situations: If agents make decision-

s on the basis imperfect information about other agents' goals, and incorporate expectations on how its decision will affect other agents' goals, then how will the evolution of cooperation proceed? How will the structure of goal (adaptive) of each agent should be self-modified for the evolution of social cooperation?

We describe the model of the interactions leading to a coordinated socially optimal behavior. It is shown the social competence behaviors can be emerged from mutual interactions based on the strategic behavior of each rational agent. The unified model of the mutual interactions among rational agents is developed, and in this model, the socially optimal behavior can be emerged as the meta-rule of each rational agent through the interactions of individuals rational agents of selfish interests.

## 2 Formulation of Competitive Interactions in an Organization

In this section, we formulate the interaction among economic agents as interdependent decision-making problems, and we especially address the question of how economic agents as economic agents infer their strategic behaviors in order to achieve their independent goals. We consider a society of economic agents,  $G = \{A_i, i = 1, 2, \dots, n\}$ . In an interdependent economic, each economic agent cannot simply proceed to perform its decision without considering what other agents may behave. A strategy of each agent should be determined solely by how its decision affects other members in an organization and how the decisions of other agents affect its own goal. We represent the adaptive function of each agent  $A_i \in G$  as

$$U_i(x_1, \dots, x_i, \dots, x_n) = x_i P_i\{x_i, x(i)\} \quad (1)$$

where  $x_i$  represents the activity level and  $P_i\{x_i, x(i)\}$  represents the social price associated to the activity of Agent  $A_i, i = 1, 2, \dots, n$ . As a specific example, we

consider the following social price scheme for each agent

$$P_i = a_i - \sum_{j=1}^n b_{ij} x_j \quad (2)$$

where  $a_i, b_{ij}, i, j = 1, 2, \dots, n$ , are some positive constant. We also denote the derivative of the adaptive function as follows:

$$M_i\{x_i, x(i)\} = \partial U_i\{x_i, x(i)\} / \partial x_i \quad (3)$$

### 3 Learning of Each individual's Adaptive Function

In this section, we provide the dynamic adaptive process in which each agent learns its adaptive function. Since the marginal utility of each agent provides the information for the direction for improving its utility, we describe the dynamic adaptive behaviors of agents as follows:

$$\begin{aligned} \text{if } M_i\{x_i(t), x(i, t)\} > 0 \quad & x_i(t+1) := x_i(t) + \delta x_i \\ \text{if } M_i\{x_i(t), x(i, t)\} < 0 \quad & x_i(t+1) := x_i(t) - \delta x_i \end{aligned} \quad (4)$$

We can describe the dynamic process in (4) by specifying the size of the modification as follows:

$$x_i(t+1) = (\alpha_i / b_{ii}) P_i(t) + (1 - \alpha_i) x_i(t) \quad (5)$$

In the equation in (5), we define the size of  $\alpha_i$  as the adaptive speed of each agent  $A_i \in G$ . We classify agents into several types depending their adaptive speeds.

We provide the decentralized computation in which each agent does not consider the other agents' behaviors, and he only needs to care the current market price. In a society consists of many agents, and without complete knowledge of other agents, agent needs to infer the strategies, knowledge, plans of other agents. Market agents can put forward their private knowledge for consideration by other agents based on its own local interactions, and agents would require the exchange of behaviors with other agents. Without communication, market agent needs infers other agents' behaviors. At the mutual interaction stage, agents can put forward their learnt knowledge for consideration by agents based on its own local interactions. This adaptive process generates a partial strategy that governs the actions of agents.

We address the question of how a society of the agents with different internal model can achieve complex collective behaviors as a whole. Different agents

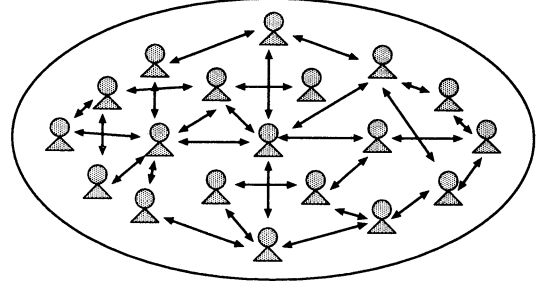


Figure 1: A society of agents with a flat organization

necessarily have not only different sets of goals or motivations, but they also have different type of internal models or cognitive states by virtue of their different histories. As a result, it is shown that the collective behavior may vary with the different types of agents. We especially address the following questions: How will the internal model of each agent affect the evolution of their collective behaviors, how will the collective behavior of agents proceed by changing the combination patterns of different types of agents' characteristics?

It is shown that the

The competitive solution is also given as the solution of the system of linear equations:

$$(B + B_1)x^0 = a \quad (6)$$

where  $B$  is a  $n \times n$  matrix with the  $(i, j)$ th element is,  $b_{ij}, i, j = 1, 2, \dots, n$ ,  $B_1$  is a diagonal matrix with the  $i$ -th diagonal element is  $b_{ii}$  and  $a, c$  are the column vectors with the elements,  $a_i, i = 1, 2, \dots, n$ , respectively.

### 4 The Evolution in an Organization

In this section, we investigate why and how rational agents form their organization. They may form organization because of the joint interest for efficient resource acquisition or allocation. We show their organization can be emerged through competitive interactions motivated by self-interested agents. We consider the two types of organization, the flat organization as shown in Fig.1 and the hierarchical organization in which  $k \times n$  agents coexist as shown in Fig.2. With the flat organization, the collective behavior at competitive equilibrium is given as

$$\begin{aligned} X_1^0 &= \sum_{i=1}^{kn} x_i^0(kn) = akn/b(kn+1) \\ G_1^0 &\equiv \sum_{i=1}^{kn} g_i^0(kn) = a^2 kn/b(kn+1)^2 \end{aligned} \quad (7)$$

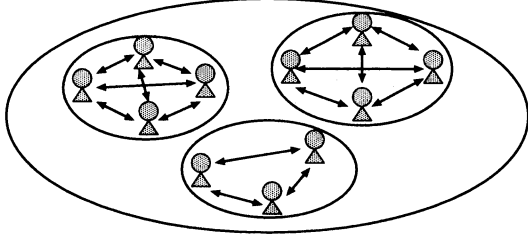


Figure 2: A society of agents with a hierarchical organization with nesting

The collective behavior at cooperative equilibrium is given as

$$\begin{aligned} X^* &= \sum_{i=1}^{kn} x_i^*(nk) = a/2b \\ G^*(nk) &\equiv \sum_{i=1}^{kn} g_i^*(nk) = a^2/4b \end{aligned} \quad (8)$$

We now consider the case of the hierarchical organization with  $k \times n$  agents with  $n$  subgroups. We assume each agent in the  $i$ -th subgroup each of them has  $k$  agents as a member of each subgroup. In Fig.2 we define the utility function as the product of the strategy, the activation level, and the price scheme. We assume agents in the  $i$ -th subgroup compete each other with the following price scheme,

$$P_i = a_i - b_i \sum_{j=1}^n x_j \quad i = 1, 2, \dots, n. \quad (9)$$

The collective behavior of the  $i$ -th subgroup at competitive equilibrium is given as

$$\begin{aligned} X_i &= a_i k / b_i (k+1) \\ G_i &= a_i^2 k / b_i (k+1)^2 \end{aligned} \quad (10)$$

The collective behavior at cooperative equilibrium is given as

$$\begin{aligned} X_i^o &= a_i k n / b_i (k+1)^2 \\ G_i &= a_i^2 k / b_i (k+1)^2 \end{aligned} \quad (11)$$

We compare the collective behavior of the flat organization and the hierarchical organization. Their ratio is given as

$$\begin{aligned} O(X_1^o/X^o) &= \{(a_i/b_i)/(a/b)\}n \\ O(G_1^o/G^o) &= \{(a_i^2/b_i)/(a^2/b)\}k^2 \end{aligned} \quad (12)$$

The summation of the all strategies  $X$  represents a kind of the activation level of the organization, therefore we now set the parameters in (9) such the activation levels of the flat organization and the hierarchical organization are the same. Then we have

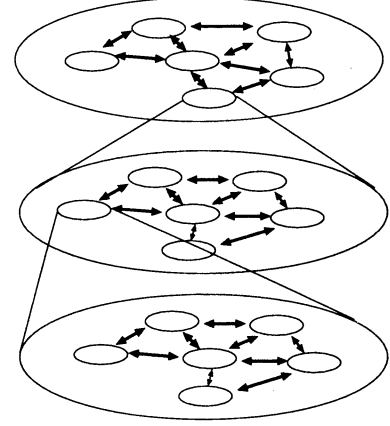


Figure 3: An organized society of agents with nesting structure

the following relations of the parameters in the price scheme defined in (9)

$$a_i = a, \quad b_i = b(kn+1)/(k+1) \quad (13)$$

Therefore their ratio is given as

$$O(G_1^o/G^o) = k^2/n \quad (14)$$

Each agent receives higher utility by forming hierarchical organization. They may form hierarchical organization because of the joint interest for efficient resource allocation and the selfish agents benefit from the hierarchical organization with the nesting structure where they can improve their own objects. We now extend the above analysis to the general hierarchical organization with the nesting structure of the level of  $s$  as shown in Fig.3. The ratio of the utility of the hierarchical organization with the multi-level nesting to that of the flat organization is then given as

$$O(G_1^o/G^o) = k^s \quad (15)$$

## 5 Conclusion

The goal of the research is to understand the types of simple local interactions which produce complex and purposive group behaviors. We formulated and analyzed the social learning process of independent economic agents. We showed that cooperative behaviors can be realized through purposive local interactions based on each individual goal-seeking. Each economic agent does not need to express its objective

or adaptive function, nor to have a priory knowledge of those of others. Each economic agent adapts its action both to the actions of other economic agents, and thus allowing previously unknown economic agents to be easily brought together to customize a group responsible for a specific mission.

## References

- [1] Barto, A.G., Sutton, " Learning and sequential decision-making", *Learning and Computational Neuro Science*, MIT Press, 1991b
- [2] Carley, K., & Prietula, M.: *Computational Organization Theory*, Lawrence Erlbaum, 1994
- [3] Marschak, J., & Radner, R.: *Theory of Teams*, Yale Univ. Press, 1972.
- [4] Ordeshook, P.: *Game Theory and Political Theory*, Cambridge Univ. Press, 1987.
- [5] Shoham, Y.: Agent-oriented Programming, *Artificial Intelligence*, Vol.60, pp51–92, 1993,
- [6] Tenney, R., & Sandell, N.R., Strategies for Distributed Decision making, *IEEE Trans. Automatic Control*, Vol.AC-19,, pp.236–247, 1974.
- [7] Young, P., The evolution of conventions, *Econometrica*, Vol.61, No.1, pp.57–84, 1993.



## An Evolutionary Design of Commitment Networks

Kimitaka Uno & Akira Namatame

Dept. of Computer Science

National Defense Academy

Yokosuka, 239-8686, JAPAN

E-mail: {kimi, nama}@cc.nda.ac.jp

### Abstract

The purpose of this paper is to study of evolutionary design of social rule. The emerged social rule is designed as the commitment network. We consider the N-persons prisoner's dilemma (NIPD) game. There are growing literatures on the bounded rationality and the evolutionary approach, the hypotheses employed in these reseaches reflect limited ability of each player or agent to receive, decide, and act upon information they get in the course of interactions. Our model can be interpreted in like manner, however, we intend to combine the evolutionary approach and the concept of mimicry and local interaction. We consider the situation where a group of agents is repeatedly matched to play a game. Each agents only interacts with his neighbors, and when agents react, they react myopically (the myopia hypothesis). Agents are completely naive and do not perform optimization calculations. Rather, agents sometimes observe the current performance of other agents, and simply mimic the most successful strategy.

**Keyword:** commitment, local interaction, mimicry, NIPD, social rule

### 1. Introduction

The iterated prisoner's dilemma game have been attracted strong interests of many researchers in diverse areas<sup>1</sup>. Many works have been towards to investigate the exact conditions that would be needed to foster cooperation on these terms. This led us to an evolutionary perspective: a consideration of how cooperation can emerge among egoists without central authority. The evolutionary perspective suggested three distinct questions. First, how can a potentially cooperative strategy get an initial foothold in an environment which is predominantly noncooperative? Second, what type of strategy can thrive in a variegated environment composed of other individuals using a wide diversity of more or less sophisticated strategies? Third, under what conditions can such a strategy, once fully established among a group of people, resist invasion by a less copulative strategy?

The purpose of this paper is to explain the problem of local interaction, mimicry and evolution in the NIPD game. We formalize these ideas in a model with a finite population of agents in which agents are repeatedly matched within a period to play a stage game. We only

impose a weak monotonicity condition reflecting the inertia and myopia hypotheses on the dynamics, which describe the intertemporal changes in the number of agents playing each strategy. The hypotheses we employ here reflect limited ability (on the agent's part) to receive, decide, and act upon information they get in the course of interactions. Our specification of dynamics draws heavily on the biological literature. In that literature, agents are viewed as being genetically coded with a strategy and selection pressure favors animals which are fitter (i.e., whose strategy yields a higher reproductive fitness or payoff against the population<sup>2</sup>). In particular, we consider the situation where a group of agents is repeatedly matched to play a game. The following three hypotheses form the basis of our analysis<sup>3</sup>.

(1) Each agents only interacts with his neighbors.

(2) When agents react, they react myopically (the myopia hypothesis).

The second interpretation is that agents are completely naive and do not perform optimization calculations<sup>4</sup>. Rather, agents sometimes observe the current performance of other agents, and simply mimic the most successful strategy. Note that in the first interpretation, agents are able to calculate best replies and learn the strategy distribution of play in society. In the second interpretation, players are less sophisticated in that they do not know how to calculate best replies and are using other agent's successful strategies as guides for their own choices.

### 2. The Problem of Commitment

The commitment problem is formulated using the payoff in Table 1. Table 1 shows the payoffs in 2IPD Game for all the possible combinations of decisions.

	Cooperate	Defect
Cooperate	R / R	S / T
Defect	T / S	P / P

Table 1. Payoff matrix in 2IPD

$$(1) T > R > P > S, (2) 2R > T + S \quad (2.1)$$

The inequality (1) in (2.1) is the condition for that if each agent seeks his individual rationality, they result in choosing (Defect, Defect). They receive  $R+R=2R$ , however if one of agent defects and the other cooperate, they receive  $T+S$ . Therefore the condition (2) in (2.1)

implies (Cooperate, Cooperate) satisfies the group rationality.

However, 2IPD cannot model complex problems such as social and economic problems in real world. It is the NIPD game that has appeared as more realistic model. Table 2 shows an example of payoff function in NIPD game. The basic principle in 2IPD game is also true for NIPD game: Defect is dominant for each agent. Generally, payoff function in NIPD game should satisfy the following condition.

$$D_n > D_{n-1}, C_n > C_{n-1}, D_n > C_n, C_N > D_0 \quad (2.2)$$

no. of cooperator	0	1	...	N-1
Cooperate	$C_0$	$C_1$	...	$C_{N-1}$
Defect	$D_0$	$D_1$	...	$D_{N-1}$

Table 2. Payoff matrix in NIPD game

As an example, let consider the following NIPD game: each agent interacts all other agents, and the payoff structure for each interaction is given in Table 1. The payoff of each agent when he cooperates and defects in the case of the number of cooperative agents is  $n$  is given as follows:

$$\begin{aligned} C_n &= (n-1)R + (N-n)S \\ D_n &= nT + (N-n-1)P, \quad (N: \text{All Agents}) \end{aligned} \quad (2.3)$$

We can measure the incentive level for defection from the value of  $D_n - C_n$ , and we term this measure as the incentive level for defection from (2.3), we obtain the following relation.

$$D_n - C_n = \{(T-R) - (P-S)\}n + (P-S)N - (P-R) \quad (2.4)$$

We describe the incentive level for defection as the function of  $n$ , the number of the cooperative agents.

	Cooperate	Defect
Cooperate	$a$	$b$
Defect	$c$	$d$

Table 3. Payoff matrix of the NIPD ( $b > 1$ )

Now we use simple payoff matrix as given in Table 3 as the same approach in Ref. 3. With this modified payoffs, the payoff of each agent when he cooperates and defects given in (2.3) is given as follows:

$$\begin{aligned} C_n &= n-1, \quad D_n = nb \\ D_n - C_n &= (b-1)n + 1 \end{aligned} \quad (2.5)$$

We can observe that the increase of  $n$  (the number of cooperators) and  $b$  (the payoff for defect) also increases the incentive level of defect for each agent. This also implies that, we can conclude that all agents defect with the rational approach. In order to induce cooperation in the society and the cooperative agents can remain, we need the social rule or regulation.

### 3. An Evolutional Approach with Mimicry and Local Interaction

Nations, businesses, tribes, and birds are examples of agents which often operate mainly within certain territories. They interact much more with their neighbors than with those who are far away. Hence their success depends in large part on how well they do in their interactions with their neighbors<sup>4</sup>. But neighbors can serve another function as well. A neighbor can provide a role model. If the neighbor is doing well, the behavior of the neighbor can be imitated. In this way successful strategies can spread throughout a population, from neighbor to neighbor.

The emergence of cooperation in the NIPD game is strongly affected by the localization of interaction. Each agent interacts with the agents on all eight adjacent squares and imitates the strategy of any better performing one. To make this process amenable to analysis, it must be formalized. For illustrative purposes, consider a simple structure of territories in which the entire territory is divided up so that each agent has eight neighbors as shown in Fig 2. In each generation, each agent attains a success score measured by its average performance with its eight neighbors. Then if a agent has one or more neighbors who are more successful, the agent converts to the strategy of the most successful of them. Or picks randomly among the best in case of a tie among the most successful neighbors.

Territorial social structures have many interesting properties. One of them is that it is at least as easy for a strategy to protect itself from a takeover by a new strategy, in a territorial structure as it is in a nonterritorial structure.

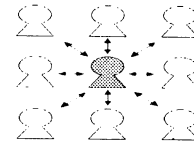


Fig 2. Localized interaction

Invasion mechanism is shown Table 4. Each square represents the payoffs of each agent at some generation  $t$  and the next generation at  $t+1$ . With the color of square white, it also represent the agent who cooperate and that of gray represents agent who defects. Generation at  $t$ , the center agent gets payoff at the level of  $8b$ , and his strategy is defect. Therefore, the condition that his neighbors may change their strategy to defect is given as follows:

$$(1) 8 < 8b \quad \text{and this implies} \quad 1 < b$$

At generation  $t+1$ , the condition that his neighbors changes to defect is also given as

$$(2) 8 < 5b \quad \text{and this implies} \quad 1.6 < b$$

We can also obtain the condition that the defect strategy invades for a diagonal direction.

$$(3) 8 < 3b \quad \text{and this implies} \quad 2.67 < b$$

With this condition of the parameter  $b$ , we can observe

that the defect strategy invades for a diagonal, vertical, right and left direction.

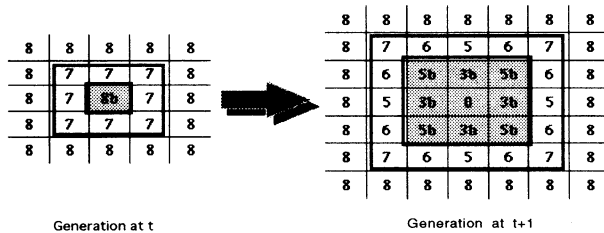


Table 4. Invasion mechanism

#### 4. How Commitment Networks will be Evolved?

In the simulation, we considered a number of agents as 2500 ( $N=2500$ ). Each cell in the figures below show that the cell with the white represents the agent with the cooperative strategy (cooperator), and the black represents the agent with the defect strategy (defector). With this NIPD game, we are especially interested in the effect of the change of the  $b$ , micromotives for defect for the collective behaviors in the whole society. We are mainly concerned with that how cooperator survive with the invasion of the defector as the change of the parameter  $b$ . Therefore, we start with only one defector in the society.

From the payoff structure in Table 3, we have the following implication.

(1)  $b < 2$ : Then the combination of (Cooperate, Cooperate) satisfies the group rationality, and the most desirable state for the society is that every agent cooperates.

(2)  $b > 2$ : The set of strategies (Cooperate, Cooperate) does not satisfy the group rationality, and the set of strategies (Cooperate, Defect), or (Defect, Cooperate) satisfies the group rationality. Then the most desirable state is that both cooperators and defectors coexist in this case.

We arranged agents for an area of  $50 \times 50$  ( $=2500$  agents,) with no a gap and observed how agents interact. Four corners and end of an area connect it with an opposite side. All agents interact eight agents in neighborhood. And only one agent in the center defects, and remaining agents cooperate at the beginning.

(1)  $b = 1.1 - 1.6$

Fig 3 shows the result of simulation. The invasion of a defector is restricted only to a neighborhood and not penetrated the whole.

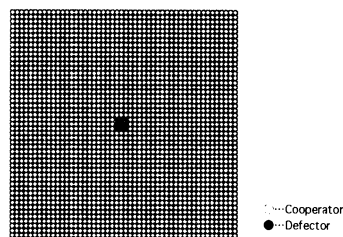


Fig 3.

(2)  $b = 1.75 - 1.99$

Only a small number of defectors invade, and the cooperator survive and become the majority in the society. In Fig 4(c), when  $b$  takes the value between 1.75 - 1.99, defectors invade only for a diagonal direction. At the 26th generation, the equilibrium is reached, and the number of defectors is 426, the number of cooperators is 2074, both cooperators and defectors coexist (as shown in Fig 4(a)). The average payoff is 0.860 (as shown Fig 4(b)).

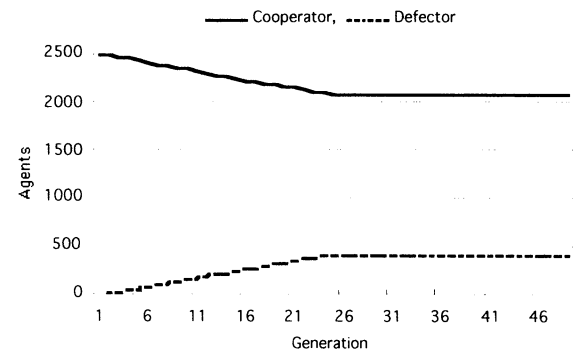


Fig 4. (a)

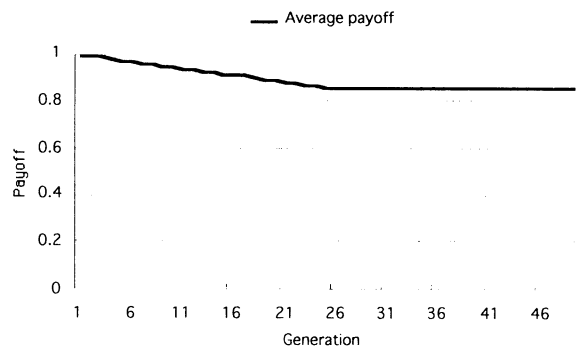


Fig 4. (b)

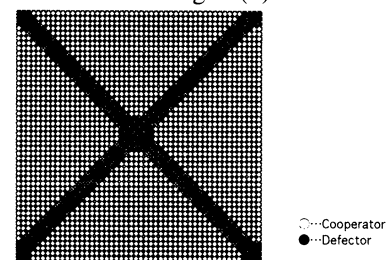


Fig 4. (c)

(3)  $b = 2.0 - 2.66$

Although defectors invade a society, cooperator can survive at level of 20%. In Fig 5(c), defector invade, however cooperator organize several colonies and defend by themselves from the invasion of defectors. As a result, at the 33th generation, at equilibrium, the number of defectors is 1962, the number of cooperators is 538, both cooperators and defectors coexist (shown Fig 5(a)). The average payoff is 0.280 (as shown Fig 5(b)).

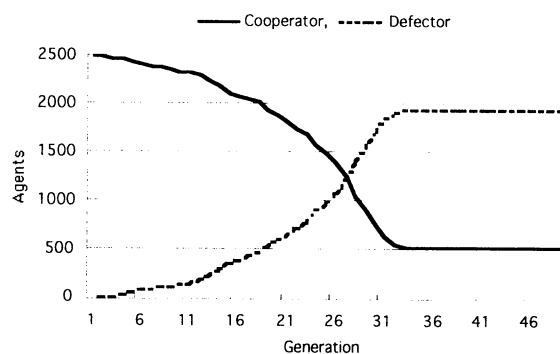


Fig 5. (a)

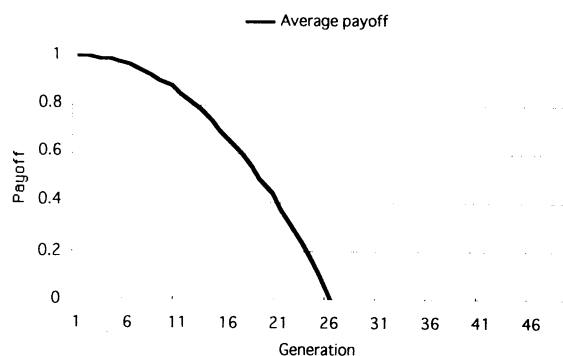


Fig 6. (b)

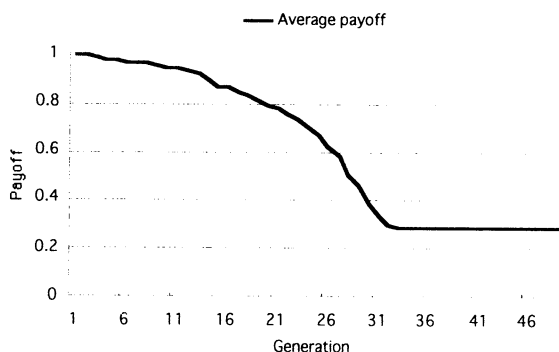


Fig 5. (b)

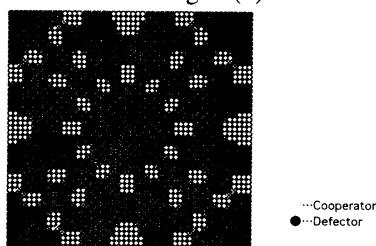


Fig 5. (c)

(4)  $b = 2.67$  over

Defectors invade a society very quickly, and the society is occupied by only defectors. When  $b$  takes the value over 2.67, defectors invade for a diagonal, vertical, right and left directions. It is shown in Fig 6(a), and at the 26th generation, at equilibrium, all agents become defectors, and the average payoff is 0 (as shown Fig 6(b)).

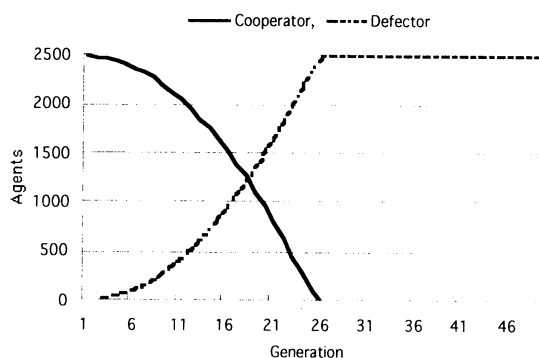


Fig 6. (a)

## 5. Conclusion

Large changes in the payoff structure can transform the interaction so that it is no longer even a Prisoner's Dilemma. If the punishment for defection is so great that cooperation is the best choice in the short run, no matter what the other agent does, then there is no longer a dilemma. The transformation of payoff does not have to be quite this drastic to be effective. However, even a relatively small transformation of the payoffs might help make cooperation based on reciprocity stable, despite the fact that the interaction is still a Prisoner's Dilemma. The reason is that the conditions for stability of cooperation are reflected in the relationship between the two outcome payoffs 1 and  $b$  in Table 3. If the payoffs change, the situation may change from one in which cooperation is not stable to one in which it is. So, to promote cooperation through modification of the payoffs, it is not necessary to go so far as to eliminate the tension between the short-run incentive to defect and the longer-run incentive to achieve mutual cooperation. It is only necessary to make the change micromotive of each agent and make the short-term incentive for mutual cooperation greater than the short-term incentive for defection.

## 6. References

1. Lloyd A. L.(1995), Computing Bouts of the Prisoner's Dilemma, Scientific American, June
2. Axelrod, R.(1984), The Evolution of Cooperation, Basic Books, New York
3. Nowak M. A., May. R. and Sigmund K.(1995), The Arithmetics of Mutual Help, Scientific American, June
4. Schelling T. C.(1978), Micromotives and Macrobbehavior, New York

## *The Evolutionary Approach for Designing a Virtual Organization*

*Yoshitugu Shimoyama, Akira Namatame*  
*Dept. of Computer Science*  
*National Defense Academy*  
*Yokosuka, 239-8686, JAPAN*  
*Tel : 0468-41-3810 (ext. 2432)*  
*Fax : 0468-44-5911*  
*E-mail: zeke@cc.nda.ac.jp*

### **Abstract**

In this paper, we provide a methodology of building collaborative working environments where, people at physically distributed location, can work together to achieve the same organizational task goals. We especially discuss the concept of emotional intelligence and the role of emotion for creative works and better cooperation. The virtual organization is defined as various members of the task forces or teams can work together over the networks. We develop the prototype model that provides the virtual organization that provides for users to explore, to work with, and to discover.

Key words: mirror agent, cooperative works, emotion, personalized knowledge environment,

### **1. Introduction**

As the tasks in our organization grow in complexity, the growing interests have been given toward methodologies that allow for efficient coordination among each member in the same organization[1][2][5]. To take advantage of the growing internets, for instance, the ways must be found to process organizational knowledge beyond specific problem domains and they must access other members of the organization. The integration of heterogeneous knowledge or view points is crucial in utilizing relevant knowledge resources distributed in an organization [3][6][7].

The virtual organization provides a knowledge world for each member of the organization to explore, to work with, and to discover. The challenge for the virtual organization is also to provide a rich working environment of a variety of subject domains that is rich enough to draw many members to use it, detailed enough to provide substantial real-world working experiences.

### **2 Concept of Emotional Intelligence**

Our concern here will be with the role of emotions as guilt, anger, envy, and even love. These emotions often predispose us to behave in ways that are contrary to our marrow it to be, others must have some way of discovering we have these emotional commitments. We will use the term commitment model as shorthand for the notion that seemingly irrational behavior is sometimes explained by emotional predispositions that help solve commitment problems. The critical assumption behind the commitment model, again, is that people can make reasonable inferences about character traits in others. By "reasonable inference" I do not mean that it is necessary to be able to predict other people's emotional predispositions with certainty just as a weather forecast. The rational model assumes certain tastes and constraints, and then calculates what actions will best serve those tastes. Widely used by military strategists,

social scientists, game theorists, philosophers, and others, it influences decisions that affect all of us. In its standard form, it assumes purely self interested tastes; namely, for present and

### **3 The Role of Emotion for Efficient Coordination**

In an organizational activities, two types of activities may occur: all members can learn as an organization, while at the same time, each member can learn by adjusting his views and actions. For the individual, he can learn to improve his problem-solving skills based on his own observation and experiences. At the cooperative stage, they can put forward their learnt knowledge for consideration by others using their own local experiences. As a group, efficient cooperative works takes effects in the form of better coordination. Cooperation would require the exchange of knowledge held by agents. The group of learning agents can offer something not available in the individuals. As a group, learning takes effect in the form of (1) better coordination (2) more efficient task and resource allocation. The improved coordination can be achieved by knowledge sharing. The task and resource allocation process can be improved by learning the specialization of learning agents by learning the group characteristics, by learning the patterns of tasks and by learning environmental characteristics. The improved coordination can be achieved by knowledge sharing or more efficient communication among members. Problem solving in an organization is a dynamic process and the actions of each member must be coordinated to achieve globally consistent and good solutions. However, the communication limits the amount of interactions among members. Thus, the coordination is always an

future consumption goods of various sorts, leisure and so on. Envy, guilt, rage, pride, love, and the like typically play no role.

important consideration, and the coordination process can be deployed in a manner parallel in which each member has a specialized representation of its characteristics and learning capability. Learning as a group can be also improved by learning the specialization of agents. An important requirement of the cooperative works, organizational decision making is also their abilities to communicate their emotion.

### **4. A Model of Mirror Agents**

An agent model specializes the framework by fixing the mental state of the agents to consist of components knowledge about the world, about themselves, about one another. The various components of the mental state and their properties are defined as follows: The mental state is determined by the past history, and the current action of agents. The actions of an agent are determined by its decisions, or choices. The agent's beliefs refer to the state of the world to the mental state of other agents..

The basic components of the mirror agent is depicted Fig. 1. It consists of the internal model and the internal memory. The internal model consists of the message processing component, the training set, the learning algorithm, and the adaptive function. The message processing component is the part of the communication board with the other mirror agents. The training set contains the specific knowledge resources to be collected, and these specifications are stored in the form of the training examples.

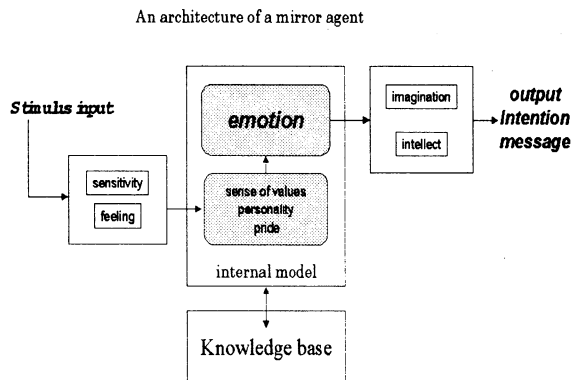


Fig. 1 : An architecture of a mirror agent

## 5. Building of the Collaborative Working Environments

The principle of cooperative works is to create a team of experts from different fields and make the team responsible for carrying out a specific task. All of the activities start at the same time, and there is a close cooperation between all the team members. Essential to the cooperative work approach, is that each team member while focusing on his area must openly share and exchange his information and results with other team members. Similarly, the team member must make sure that his work is consistent with that of other team members' work. Cooperative work therefore, creates a dynamic, interconnected network of knowledge among the team members. Cooperative work, if it is by a team of engineers, or by a group of members, requires coordination, communication, knowledge sharing and sharing their emotion [2][4][7]. The computing environment to support such teams is an open, interactive, distributed environment. For information management, each member, needs to have a large degree of autonomy in structuring and organizing his own information, while he has to also consider the needs of the other team members. The act of cooperation creates

dependencies that have to be negotiated, and administered. These dependencies transform in turn into obligations, which have to be respected by each team member, and which constrain the autonomy of each team member. The integration of independently developed information systems into a cooperating federation of agents poses similar problems. These information systems are heterogeneous, use different data models, inference mechanisms, and so on, but the integrated system should provide a common universe of discourse.

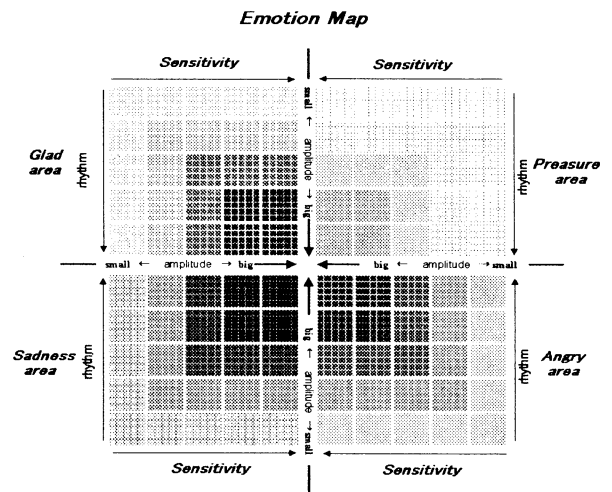
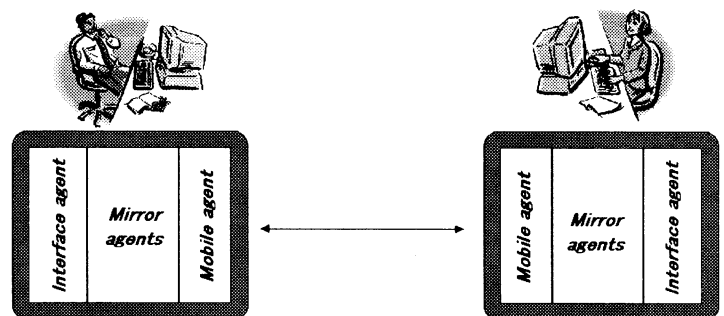


Fig.2: Emotion map(Mirror agent is a screen of user's emotion)

*Mirror agent is a mirror of user's emotion.  
Users express their emotion such like making a message.*



- Interface agent is an editor of message which contains user's emotion.
- Mirror agents support user's intention and emotion expressing.
- Mobile agent supports communication between agents.

Fig.4: Network by Mirror agents

## 6. Conclusion

This paper provided a new methodology for building personalized working environments as the virtual organization. We especially discussed the concept of emotional intelligence and the role of emotion for creative works and better cooperation. The virtual organization is defined as where various members of the task forces or teams co-work over networks. We introduced the concept of mirror agents each of which endowed with the learning capability, represents a specific problem domain and is specialized to interact with the environment. The most important characteristic of mirror agents is that they may be amended or modified. Each mirror agent is encapsulating a specific set of knowledge obtained from a different set of training examples. At the cooperative stage, mirror agents put forward their learnt knowledge. We defined the virtual organization as repositories of various and heterogeneous knowledge resources distributed over in an open network environment.

As an application, we proposed the Virtual Organizational Decision Making Environments (VODM) may provide highly structured computer-based decision-making environments.

## References

- [1] Alan, H.B. and Gasser, L., : Readings in Distributed Artificial Intelligence, Morgan Kaufmann, 1988.
- [2] Baecker, R.M., Readings in Groupware and Computer-Supported Cooperative Work, Morgan Kaufmann, 1993.
- [3] Carley, K, & Prietula, M.: Computational Organization Theory, Lawrence Erlbaum, 1994
- [4] Collet, C., Huhns, M.N. & Shen, W., " Resource Integration Using a Large Knowledge Base in Carnot, IEEE Computer, Vol. , No. 12, pp. 55-65, 1991.
- [5] Cutkosky, M., Genesereth, M., Mark. W and Tenenbaum, J.M., " PACT: An Experiment in Integrating Concurrent Engineering System," IEEE Computer, pp.28-37, Dec. 1991
- [6] Neches, R. and Fikes, R., " Enabling Technology for Knowledge Sharing", AI Magazine, Vol. 1991, pp. 37-55, 1991.
- [7] Witting, T., ARCHON: An Architecture for Multi-Agent Systems, Ellis Horward, 1992.



## What Facilitates Emergence of Symbiosis

Ming Chang    Kazuhiro Ohkura    Kanji Ueda

Faculty of Engineering  
Kobe University  
Rokkodai, Nada-Ku, 657, Japan  
email: {chang,ohkura,ueda}@mi-2.mech.kobe-u.ac.jp  
Phone:+81-78-803-1119 FAX:+81-78-803-1131

### Abstract

It is argued that symbiosis in the natural world is of great evolutionary significance, and symbiosis concept has been applied for evolutionary learning algorithm design. By considering symbiosis as an emergent phenomenon and a source of innovations in the evolutionary learning algorithm design, we expect an enhanced learning ability can be embedded in the learning algorithms. The purpose of this study is to investigate conditions for emergence of symbiotic association according to relations between evolution rates of the species and the change rate of the environment that surrounds the species. Our preliminary computer simulations show that the emergence of symbiosis is strongly dependent on the evolution rates of species and the environment change rate. Since the emergence of symbiotic association can be viewed as a solution to unpredictable environment variability and symbiosis results an integration of the genetic material, the learning algorithms which embedded symbiosis phenomenon can be more robust and express more complex behavior.

**Keywords:** Symbiosis, Fitness landscape, NKC model.

## 1 Introduction

### 1.1 Symbiosis in Biology

In the natural world, various organisms cooperate and/or compete with each other in an ever-changing environment, and they continuously surprise us with their wonderful adaptive variation. Symbiosis is one of those amazing phenomena. The term symbiosis was originally defined as meaning the “living together” of “dissimilar” or “differently named” organisms, and it is now mostly taken to mean “mutualistic biotrophic associations”. When defined it with respect to out-

come, symbiosis is “an internal partnership between two organisms (symbionts) in which the mutual advantages normally outweigh the disadvantages” [1]. Although there is no consensus of the definitions of symbiosis, we here, according to [5], consider it as a phenomenon that organisms of different species live together in a close association, resulting in a raised level of fitness for one or more of the organism.

In Darwinian evolution, the tendency, through time, toward complexification and specialization, can be identified. But natural selection can be effective only where there are certain quantitative relationships among sampling errors, selection coefficients, and rates of random change, it works only with an immediate better-vs.-worse among individuals in a population. Once a certain level of complexity is evolved, selection will maintain adaption by occasionally substituting one adaptive character for another, but this will not result in any of the kinds of cumulative progress that have been envisioned [2]. Even Charles Darwin himself recognized the insufficiency of natural selection alone to produce either new species, other new taxa, or increased morphological complexity in the living world. From the idea of *evolutionary progress*, for example, in one of the stages during evolution - transition from prokaryote to eukaryote - symbiosis played a crucial role. Symbiosis has also been considered as a source of evolutionary novelties, since it affords a mechanism whereby genetic material from very distantly related organisms can be brought together in a single descendant, resulting a potentially increased morphological complexity [3].

### 1.2 Symbiosis in Evolutionary Algorithms

Evolutionary Algorithms (EA) are computer-based problem solving systems based on the model of nat-

ural evolution. EA typically perform population-based stochastic search in the search space, and there is communications and information exchange among individuals in the population when generate new individuals[6]. In general, the populations in EA conclude individuals with the same structure, *e.g.* in the *canonical* Genetic Algorithms (GA) case, this means that individuals in the population were represented by bit-strings with the same size. For the problems with a predefined search space, such as parameter optimization[7], the fixed-structure genotype representation is well suitable[10]. But if the purposes are trying to evolve complex structures with potentially unrestricted capabilities as in the evolution of animates or cognitive structures where we are facing an infinite dimensional search space, the variable genotype structure seems favorable. For example, some biologists have suggested that the *Red Queen effect* arising from coevolutionary arm races has been a prime source of evolutionary innovations and adaptations[4]. Some coevolutionary algorithms have shown significant success[9]. But if we consider coevolution as something rather than the game of “paper, stone and scissors”[11], the variable-structure for individual is necessary. Since genetic material in different ancestral lineages is brought together by symbiosis, where each of them reflects the different features of the problem, symbiotic association will result an enhanced problem-resolving ability.

## 2 Evolutionary Rates Conditions for Emergence of Symbiosis

In this section, the proposed model is introduced, and the conditions for the emergence of symbiotic relation according to environment change rate and evolutionary rates of species are investigated.

### 2.1 Proposed Model

In our computer simulations, we assume two species(population)  $S_A$  and  $S_B$  in a global environment  $E$ . The two species are initially genetic specified and evolve independently. But during the simulation, in each generation, an offspring  $s_b$  of an individual in  $S_B$  will attach to an individual  $s_a$  in  $S_A$  stochastically. If the attachment occurred, we call the attached  $s_a$  and  $s_b$  as host and parasite, respectively. The attached host and parasite is called as *host-parasite* pair, and the host-parasite pairs compete with each other

as well as the other members in  $S_A$ . Over generations, if the frequency of host-parasite pairs in  $S_A$  become relatively static but not zero, we say that the symbiosis emerged. In the following, we describe the model more detailed.

#### 2.1.1 Represent Individual by NKC model

In our model, the individuals of both species are represented by NKC model[8] and the environment is represented by a bit string. In the NKC model,  $N$  represents the number of genes(bits) in a haploid chromosome(bit string),  $K$  and  $C$  represent the number of linkages each gene has to other genes in the same and different chromosome(s), respectively. For our purpose here, each gene has  $W$  linkages to the environment further. To calculate the fitness of the entire chromosome, the fitness contribution from each locus is averaged as follows:

$$f(chromosome) = \frac{1}{N} \sum_{i=1}^N f(locus_i),$$

where the fitness contribution of each locus,  $f(locus_i)$ , is determined by the value of gene  $i$  together with values of the  $K$  intracting genes in the same chromosome,  $C$  interacting genes from different chromosomes and  $W$  bits of the environmental bit string. The values of those genes were used as an index into a table  $T_i$  of size  $2^{K+C+W+1}$  of randomly generated numbers uniformly distributed over the interval  $[0.0, 1.0]$ . For a given gene  $i$ , the set of  $K$ ,  $W$  linked bits were randomly selected from the same chromosome and environmental bit string, respectively. Because it is more natural to envision that the fitness of an individual is affected by only a subset of all individuals in the same species rather than all of the species, for each gene  $i$ ,  $C$  interacting genes were selected from the same individual in the same species.

In each generation, individuals of each species perform random hill-climbing search. First each individual of  $S_A$  and  $S_B$  inverts the value of its every gene with probability  $A_m$  and  $B_m$ .  $A_m$  and  $B_m$  was said as their *evolution rates*. After the inversion, if fitness of the individual was enhanced, the inversion was accepted, otherwise the individual will invert back to the original value.

#### 2.1.2 Here, Symbiosis means Attachment

After random hill-climbing search of each species was over, an individual in  $S_B$  is randomly selected to produce a copy of it, and we name the copy as the offspring  $s_b$  of the individual. The fitness of  $s_b$  is the

same as its parent. At the same time, an individual is also randomly selected from  $S_A$ , we assume that if the fitness of  $s_b$  is greater than a randomly generated number from  $[0.0, 1.0]$ , the attachment between  $s_b$  and  $s_a$  was successful, and the horizontal genetic material transmission occurred. By “horizontal” we means that different from the genetic material is organized and transitions between generations, this time it takes place within the generations.

After the attachment, the pair of  $s_a$  and  $s_b$  was called as *host-parasite* pair. For Brevity, we further assumed that every individual in  $S_A$  can be attached by only one “ $s_b$ ”, furthermore, because  $s_b$  is not in  $S_A$ , the C interacting genes of it were set unchanged in further generations, and W linkages to the environment of the  $s_a$  now is selected from the host  $s_a$ , but not from  $E$  any more. The fitness resulted host-parasite pairs is the average of the genes of  $s_a$  and  $s_b$ . The host-parasite pairs must compete with each other as well as other unattached individuals of  $S_A$  for survival. In each generation, all the members of  $S_A$  were ranked according to their fitness, for members with lower fitness, if they were attached by parasite, only the parasite was killed, else nothing was done. By doing so, we can concentrate on the frequency of the host-parasite pairs in  $S_A$ .

In each generation, the environmental bit string inverts each bit value with probability  $W_m$ , and  $W_m$  was the environment change rate.

### 3 Simulation Results

In our simulations, each species has 50 individuals. For each individual in both species,  $N = 50, K = 2, C = 2, W = 2$ , respectively. The environmental bit string has 50 bits. In each generation, from the five lowest-fitness members in  $S_A$ , the parasite  $s_b$ 's were deleted. Frequency of host-parasite pairs is averaged over 3000 generations, and the last results are averaged over 10 runs.

Figure 1 show that when the environment change rate was set unchanged at 0.01, the frequency of the host-parasite pairs will increase with lower  $A_m$  while not affected by  $B_m$  (upper), and the host-parasite pairs increase along with the environment change rate (middle). The lower figure shows that there is a land subsidence occurred where  $A_m < B_m$ . Those phenomena can be explained as before attachment,  $s_b$  is affected by the environment because its W linkage genes were selected from environment bit string, but after the attachment, the W linkage genes were selected from the host  $s_a$ , so the host is now the new

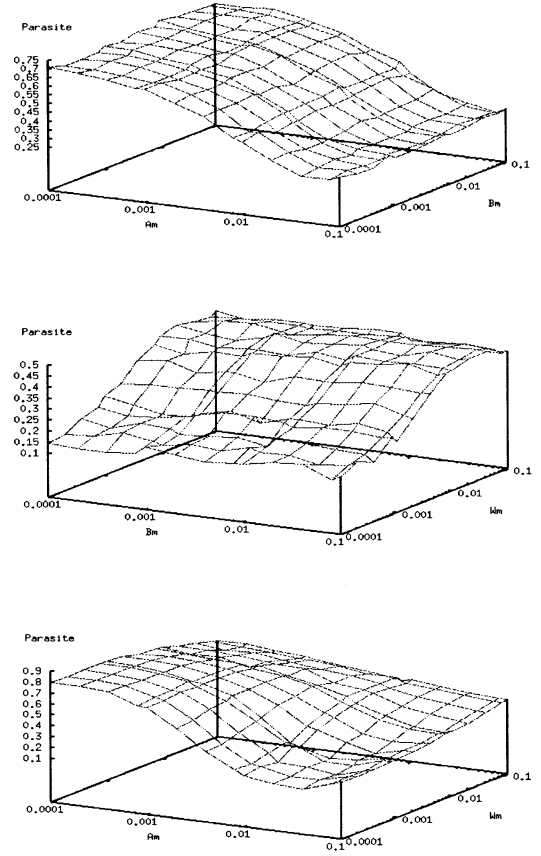


Figure 1: Frequency of host-parasite pairs when the environment change rate ( $W_m$ , upper), evolutionary rate of  $S_A$  ( $A_m$ , middle) and  $S_B$  ( $B_m$ , lower) are set at 0.01 unchanged during the simulation, respectively.

“environment” for the parasite. If the environment change too fast, the fitness of both species will decrease. When the environment change rate is higher than the evolutionary rate of  $S_A$ , for the members of  $S_B$ , attach to individuals in  $S_A$  will result a relatively stable “environment”, so the symbiosis, here we mean the host-parasite relation, emerged.

In Figure 2, where only the individual with greater fitness than the average fitness of parasites now attaching on individuals of  $S_A$  will be permitted to attach, we can see that the frequency is also affected by the change on  $B_m$ .

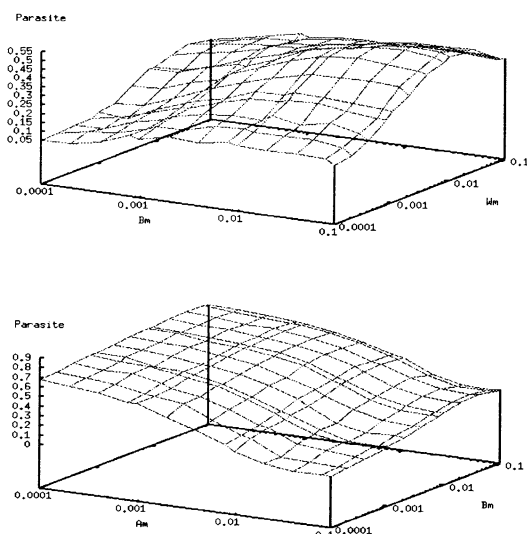


Figure 2: Only  $s_b$  with greater fitness than the average fitness of parasites is permitted to attach while the evolutionary rate of  $S_A$  (upper) and the environment change rate  $W_m$  (lower) are set at 0.01 unchanged during the simulation.

## 4 Summary

In this paper, we discussed the conditions for emergence of symbiotic association according to relations between environment change rate and evolution rates of the species under the same environment. The simulation results show that the emergence of symbiosis is strongly dependent on the evolution rates of species and the environment change rate. For example, when the environment changes is too fast for the species to adapt, it is advantage for the species to find an comparative stable environment by attaching to other species.

## Acknowledgment

This research is partially done under the financial support through "Methodology for Emergent Synthesis" project (project number: 96P00702) in Research for the Future Program of the Japan Society for the Promotion of Science (JSPS). The authors acknowledge that the discussion with other project members was directive for this research.

## References

- [1] Lynn Margulis, "Symbiogenesis and Symbiontism," In L. Margulis & R. Fester (Eds.) *Symbiosis as a source of evolutionary innovation*, pp. 1-14, Cambridge, MA: MIT Press, 1991.
- [2] George C. Williams, *Adaptation and Natural Selection*, Princeton University Press, 1974.
- [3] John Maynard Smith, "A Darwinian View of Symbiosis," In L. Margulis & R. Fester (Eds.) *Symbiosis as a source of evolutionary innovation*, pp. 26-39, Cambridge, MA: MIT Press, 1991.
- [4] R. Dawkins, *The Blind Watchmaker*, Longman, Essex, 1986.
- [5] Lawrence Bull and Terence C. Fogarty, "Artificial Symbiogenesis," *Artificial Life* Vol.2, No. 3, pp. 269-292, 1995.
- [6] X. Yao, "An Overview of Evolutionary of Computation," In *Chinese Journal of Advanced Software Research*, Vol. 3, No. 1, pp. 12-29, allerton Press, Inc., New York, NY 10011, 1996.
- [7] Thomas Bäck and Hans-Paul Schwefel, "An Overview of Evolutionary Algorithms for Parameter Optimization," *Evolutionary Computation*, Vol. 1, No. 1, pp. 1-23, 1993.
- [8] S. A. Kauffman, *The Origins of Order*, Oxford University Press, 1993.
- [9] D.W. Hills, "Co-evolving parasites improve simulated evolution as an optimization procedure," In C.G. Langton, C. Taylor, J.D. Farmer, and S. Rasmussen, (Eds.) *Artificial Life II*, pp. 313-324, Addison-Wesley, 1990.
- [10] Mitchll A. Potter and Kenneth A. De Jong, "A Cooperative Coevolutionary Approach to Function Optimization," In *The Third Parallel Problem Solving From Nature*, Jerusalem, Israel, pp. 249-257, Springer-Verlag, 1994.
- [11] D. Cliff and G.F. Miller, "Tracking the Red Queen: Measurements of adaptive progress in co-evolutionary simulations," In F. Morán, A. Moreno, J.J. Merelo, and P. Chacón, (Eds.) *Advances in Artificial Life: Proc. Third Euro. Conf. Artificial Life*, pp. 200-218, Springer-verlag, 1995.

## Synthetic Collective Behavior by Multiple Reinforcement Learning Agents in Simulated Dodgeball Game

N.ONO and S.YOSHIDA

Dept. of Information Science and Intelligent Systems  
Faculty of Engineering, University of Tokushima  
2-1 Minami-Josanjima, Tokushima 770-8506, JAPAN

### Abstract

Several attempts have been reported to let multiple monolithic reinforcement-learning agents synthesize collective decision policies needed to accomplish their common goal effectively. Most of these straightforward reinforcement-learning approaches, however, scale poorly to more complex multi-agent learning problems, because the state space for each learning agent grows exponentially in the number of its partner agents engaged in the joint task. To remedy the exponentially large state space in multi-agent reinforcement learning, we previously proposed a modular approach and demonstrated its effectiveness through the application to the pursuit problem. In this paper, the effectiveness of the proposed idea is further demonstrated using yet another multi-agent learning problem inspired by dodgeball game.

## 1 Introduction

In attempting to let simple reactive agents synthesize some collective behavior, many researchers in the fields of artificial life and machine learning have applied monolithic reinforcement learning (RL) algorithms to multi-agent learning problems. In most of these applications, only a small number of learning agents are engaged in their joint tasks and accordingly the state space for each agent is relatively small. This is the reason why monolithic RL algorithms have been successfully applied to these multi-agent learning problems. However, these straightforward applications of RL algorithms scale poorly to more complex multi-agent learning problems, where not a few learning agents are engaged in some coordinated tasks[4]. In such a multi-agent problem domain, agents have to learn to appropriately behave, each perceiving not only sensory information produced by the physical environment itself but also that produced by the other

agents, and hence the state space for each RL agent grows exponentially in the number of agents operating in the same environment. Even simple multi-agent learning problems are computationally intractable by the monolithic RL approaches.

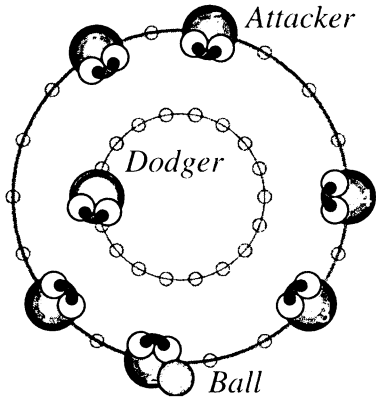
Previously, to remedy the problem of combinatorial explosion in multi-agent RL, we proposed a modular approach in [2], based on Whitehead's idea[6]. We considered the pursuit problem[1] as a multi-agent learning problem suffering from the combinatorial explosion, and showed how successfully modular RL prey-pursuing agents synthesize coordinated decision policies needed to capture a randomly-evading prey agent[2, 3]. In this paper, we attempt to additionally validate the effectiveness of our modular approach through the application to yet another multi-agent learning problem, called the *Simulated Dodgeball Game-II*.

## 2 Problem Domain

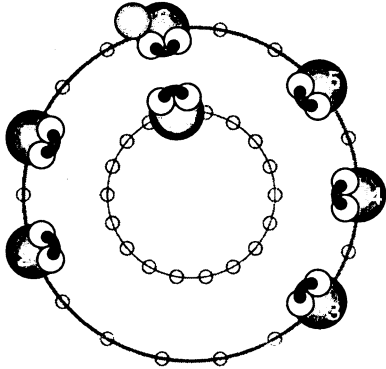
We consider the *Simulated Dodgeball Game-II* (*SDG-II*), summarized as follows:

- A single *dodger* agent and multiple *attacker* agents are operating in the discretized game space as shown in Fig. 1 (a). The attackers move on the outer circle, and their common objective is to hit the dodger by a ball. The dodger moves on the inner circle, and its objective is to survive all the shots made by the attackers.
- Initially, all the agents are placed at random positions on their corresponding circle, and a single ball is given to a randomly-chosen attacker. At each time step, they perceive the current situation in the world, and then select and perform their own action without communicating with each other.

- At each time step, the dodger and attackers can make a step to their left or right, and they can remain at their current position. In addition, the ball-holding attacker can make a pass (i.e., pass the ball) to one of the other attackers, and make a shot against the dodger. An attempt to make a pass or shot does not always succeed. A pass succeeds if and only if the passer and receiver are located on the adjacent positions. A shot succeeds if and only if the shooter and the dodger exactly face each other, as shown in Fig. 1 (b), for example.



(a) Typical Configuration.



(b) Precondition for making a successful shot.

Figure 1: The *Simulated Dodgeball Game-II*.

We suppose that the dodger’s behavior is manually-programmed in advance, and it simply moves so as to maximize the distance to the current ball-holding attacker. It means that without making a pass, the

ball-holder can not reduce the distance to the dodger<sup>1</sup>. Hence, to hit the dodger by the ball, the attackers have to come close to the dodger by making a pass repeatedly until the distance between the dodger and the current ball-holding attacker is minimized, and finally the ball-holder has to make a shot against the dodger. The question is whether the attackers can synthesize such collective behavior, based on some RL algorithm.

### 3 Implementation

In *SDG-II*, each attacker’s state space is huge, since it perceives the current ball-holding attacker’s identifier, the relative position to the dodger, that to each of the other attackers. To reduce its state space size and speed-up the learning, we implemented each attacker agent based on the variant of modular Q-learning architecture[2].

This architecture consists of multiple learning-modules and a single mediator-module. Each learning-module focuses on specific attributes of the current state and performs Q-learning. More specifically, each module observes the current ball-holder’s identifier, the relative position to the dodger, and that to two specific attackers only, and it does not care any other attackers. Note that each module’s state space size is kept independent of the number of attackers involved in the game, and hence it does not grows exponentially in the number.

On the other hand, a mediator-module, without learning, combines learning-modules’ decision policies using a simple heuristic decision procedure and determines a final decision by the corresponding agent. In our preliminary experiments, the mediator-module selects the following action:

$$\arg \max_{a \in A} \sum_k Q^{i_k}(x_{i_k}, a)$$

where  $Q^{i_k}$  denotes the action-value function implemented by the learning-module  $i_k$ .

### 4 Simulation Results

Our simulation run consists of a series of trials. A trial begins with a single dodger agent and multiple attacker agents placed at random positions, and it ends when (i) a shot is made, irrespective of its results, (ii)

<sup>1</sup>To reduce the distance, the receiver has to make a step when catching the ball.

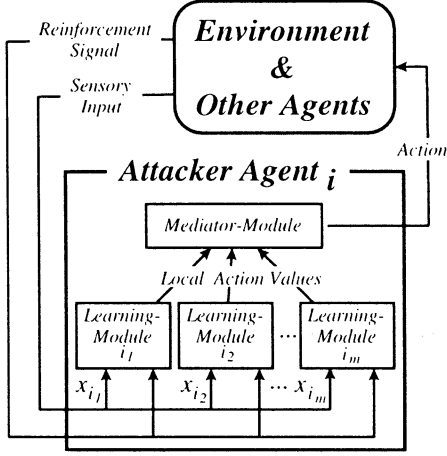


Figure 2: Modular RL architecture for an attacker agent.

a pass is made but it fails, or (iii) the dodger is not hit within 100 time steps. Then, individual agents are relocated at random positions.

Immediately after performing an action, the attackers receive the following reinforcement signals:

- when a shot is made successfully, the shooter and the other attackers immediately receive rewards of 1.0 and 0.5, respectively<sup>2</sup>,
- when a shot is made but it fails, the shooter receives a penalty of  $-0.5$ , and all the others receive a cost of  $-0.1$ ,
- when a pass is made but it fails, the passer receives a penalty of  $-0.5$ , and the other attackers receive a cost of  $-0.1$  uniformly;
- otherwise, the attackers receive a cost of  $-0.1$ .

The learning rate  $\alpha$  and discount factor  $\gamma$  are set to 0.1 and 0.9, respectively. Initial  $Q$ -values for individual state-action pairs are randomly selected from the interval  $[0.01, 0.1]$ .

We employ the following settings: Five attackers and a single dodger are operating in the game environment. Inner and outer circles are uniformly divided into 12 segments, each corresponding to a discrete position. When a trial begins, the dodger is located so that it is most distant from the initial ball-holding attacker.

<sup>2</sup>When a modular reinforcement-learning agent receives a reinforcement signal, all of its component learning-modules uniformly receive the same signal.

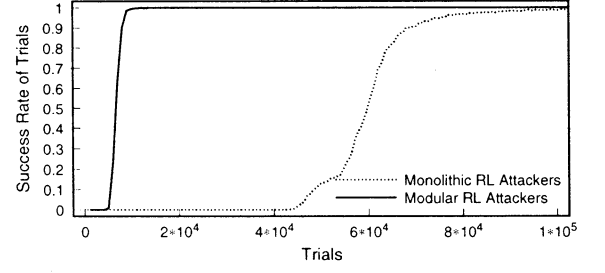


Figure 3: Average success rate of trials. Each curve is an average over 10 simulation runs.

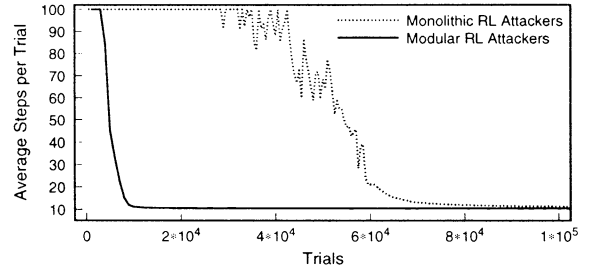


Figure 4: Average number of time steps per learning trial. Each curve is an average over 10 simulation runs.

The results of our simulation runs are shown in Fig. 3 and Fig. 4. The solid curve in Fig. 3 indicates the average success rate of trials, and that in Fig. 4 indicates the average number of time steps per trial. At initial trials, the attackers can not hit the dodger by the ball, but shortly they start improving their performance significantly. Eventually they come to steadily accomplish their common goal at every trial in any of our simulation runs.

To see the efficacy of our modular Q-learning approach compared with a straightforward monolithic version, we tried to solve the same *SDG-II* using a monolithic Q-learning approach. The dashed curves in Fig. 3 and Fig. 4 indicate the performance by the monolithic Q-learning attackers. As suggested by the figures, they are totally outperformed by the modularized attackers.

Collective behavior typically exhibited by the modular Q-learning attackers is illustrated in Fig. 5. As shown in the figure, the attackers reduce the distance between the current ball-holding attacker and the dodger incrementally by taking advantage of a chain of passes, and eventually they can have a chance

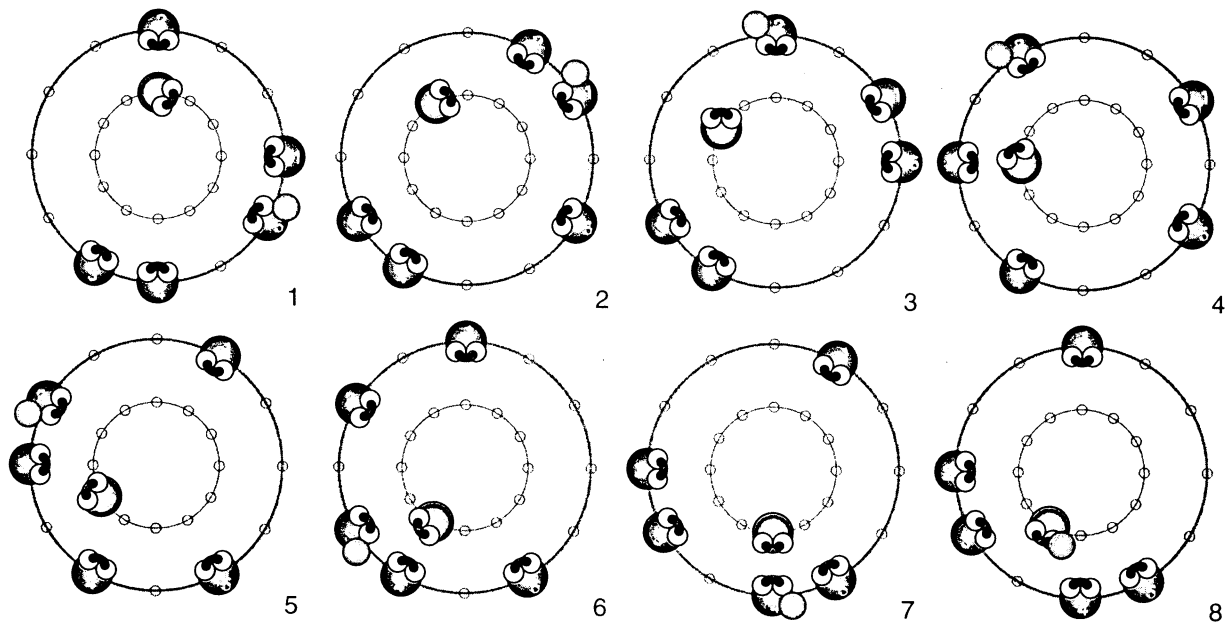


Figure 5: Typical behavior by attacker agents. Note that the distance between the ball-holding attacker and the dodger is reduced each time a pass is made.

of making a successful shot.

## 5 Concluding Remarks

To remedy the problem of combinatorial explosion in multi-agent RL, we proposed a modularized RL approach [2], inspired by Whitehead’s idea[6]. Although our approach has been successfully applied to various variants of the pursuit problem(e.g., [2, 3]), its effectiveness has not been clarified using the other learning problems.

In this paper, using yet another multi-agent learning problem, we additionally demonstrated the effectiveness of the approach. As a new illustrative problem suffering from the combinatorial explosion, we considered the Simulated Dodgeball Game-II(*SDG-II*), and showed how successfully a collection of modularized RL attacker agents synthesize coordinated behavior needed to achieve their common goal.

Multi-agent learning is a difficult problem in general, and the results we obtained strongly rely on specific attributes of the problem just as in the case[2]. But the results are quite encouraging and suggest that our modular RL approach is promising in studying adaptive behavior of multiple autonomous agents. In the next phase of this work, we intend to let multi-

ple RL agents identify its appropriate modularization according to their joint tasks.

## References

- [1] Benda, M., V.Jagannathan, and R.Dodhiawalla: On Optimal Cooperation of Knowledge Sources, Technical Report BCS-G2010-28, Boeing AI Center, 1985.
- [2] Ono, N. *et al.*: Multi-agent Reinforcement Learning: A Modular Approach, Proc. of the 2nd International Conference on Multiagent Systems, AAAI Press, 1996.
- [3] Ono, N. *et al.*: A Modular Approach to Multi-Agent Reinforcement Learning, in G.Weiß(Ed.): Distributed Artificial Intelligence Meets Machine Learning: Learning in Multi-Agent Environments, Springer-Verlag, 1997.
- [4] Tan, M.: Multi-agent Reinforcement Learning: Independent vs. Cooperative Agents, Proc. of the 10th International Conference on Machine Learning, 1993.
- [5] Watkins, C.J.C.H.: Learning With Delayed Rewards, Ph.D.thesis, Cambridge University, 1989.
- [6] Whitehead, S. *et al.*: Learning Multiple Goal Behavior via Task Decomposition and Dynamic Policy Merging, in J.H.Connell *et al.* (Eds.): Robot Learning, Kluwer Academic Press, 1993.



# Evolutionary Design of Analog Electronic Circuits

Hajime Shibata   Saed Samadi   Hiroshi Iwakura  
Dept. of Communications and Systems  
University of Electro-Communications  
Chofu, Tokyo 182-8525

## Abstract

An automatic cell-level analog circuit design method based on Genetic Algorithms is presented. This method generates circuit topology from electrical specifications, and does not require expert knowledge about circuit topology. The genotype is a layered set of grids composed of cells possessing numerical values. A method called PGA coding is proposed to translate the genotype to circuits. The phenotype evolves from initial poor performance as generations go by and circuits meeting the specifications with greater accuracy begin to appear. Automatic design simulations show the capability of this method to evolve an inverter. We also identify some practical problems of automatic circuit design based on evolutionary computation.

*Keywords* — Analog circuit synthesis, design automation, circuit description, genetic algorithms.

## 1 Introduction

A number of automatic analog circuit design methods treating sizing issues, e.g. component values or device sizes, have been proposed [1, 2]. Most of these methods use fixed circuit structures, while some of them optimize circuit topology within the limits of pre-designed sub-circuits [3]. On the other hand, Koza et.al., recently proposed an automated analog circuit synthesis method by means of Genetic Programming (GP) [4] that simultaneously generates both circuit topology and sizing using electrical specifications. Reference [4] is a pioneering work, because it showed the power of an evolutionary method in tackling a design problem considered to be highly complex needing an expert human designer. However it is not known, either experimentally or theoretically, whether GP combined with cellular coding [5] is the best way to evolve analog circuits. It is also important to consider the realization issues of the genetically generated circuit.

This paper presents an evolutionary design method of electronic circuits which only needs the specifications data be provided. It is based on a Genetic Algo-

rithm (GA)[6] with a unique genotype structure. The chromosomes are composed of cells placed on layered grids and have real numerical attributes. A method called PGA (Pin Grid Array) coding is proposed to translate the chromosome to a circuit. The generated circuits are evaluated by circuit simulator SPICE [7]. Since the circuit structure is determined by the evolutionary process working on the genes, expert knowledge is not required. Simulation results of evolving an inverter using the proposed method are presented. We also treat practical issues including the existence of junk circuits in the genetically designed circuit.

The paper is organized as follows. The topology constructing method, the PGA coding, is detailed in Section 2. Section 3 describes the overview of design process and a design example is presented in Section 4. Section 5 provides some concluding remarks and future research topics.

## 2 PGA Coding

An analog electrical circuit can be decomposed to several layers each corresponding to the connection pattern of a certain type of element. A layer is defined on a rectangular grid for each type of element  $e$ . The grid is composed of cells  $c_k$  with numerical values  $V(c_k)$ . The set  $\mathcal{C}$  represents all cells belonging to the grid. We propose an algorithm to generate elements  $e_i$  between the grid points based on  $V(c_k)$  and a Manhattan distance defined between grid cells.

Let set  $\mathcal{S}_i$  denote the grid cells that are connected through the element  $e_i$  (the  $i$ -th element of type  $e$ ). For each type of component, the PGA coding generates  $\mathcal{S}_i$  until no further elements can be added. The function  $I(\mathcal{S}_i)$  represents the intensity of potential connections between cells in  $\mathcal{S}_i$ , and is calculated as

$$I(\mathcal{S}_i) = \sum_{c_k \in \mathcal{S}_i} V(c_k) - p f_d(\mathcal{S}_i) \quad (1)$$

where,  $f_d(\mathcal{S}_i)$  is the maximum Manhattan distance between cells included in  $\mathcal{S}_i$ , and  $p$  is a positive real number called penalty coefficient. Let  $n$  denote the num-

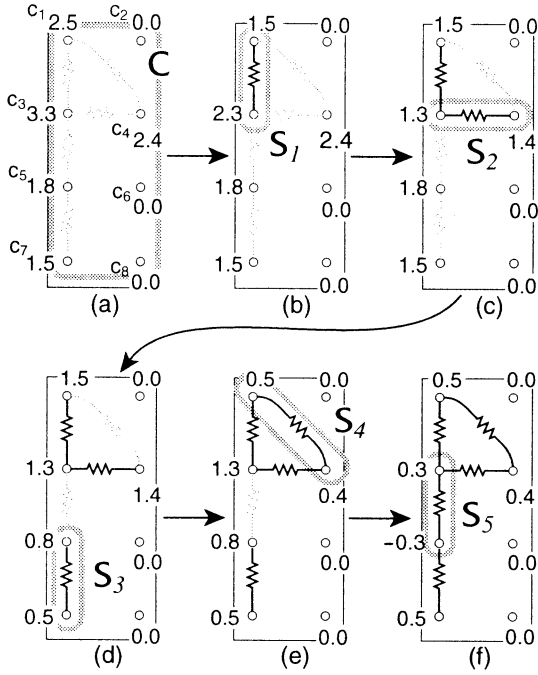


Figure 1: Generation process of elements between cells of the resistor layer using the PGA coding algorithm. The numerical values of cells are also shown.

ber of terminals of the element type corresponding to a layer. The  $n$ -terminal elements of type  $e$  are generated on the grid by executing the following algorithm.

1.  $i = 1$ .
2.  $S_i = \phi$ .
3. Select a cell  $c_f \in \mathcal{C}$  that has the maximum  $V(c_f)$  and add it to  $S_i$ .
4. While ( $\text{Card}\{S_i\} < n$ ) select a cell  $c_n$  with highest value of  $I(S_i \cup \{c_n\})$ , and  $S_i \leftarrow S_i \cup \{c_n\}$ .
5. If  $I(S_i) \geq n$  then insert an element between terminals that are represented by  $S_i$  and  $i \leftarrow i + 1$ . Else quit.
6. Subtract value 1 from cells included in  $S_i$ .
7. Goto 2.

Fig. 1 is an example of the case for 2-terminal elements. Elements like diodes and transistors with polar terminals require sub-layers to distinguish between their terminals. In this case, the set  $S_i$  is formed from cells lying in different sub-layers.

To get the overall circuit, we superimpose the elements generated in each layer on each other. Specifically, layer nodes having same grid coordinates are connected. In case of existence of elements with electrically floating terminals, such terminals are connected to the nearest terminals belonging to neighboring elements.

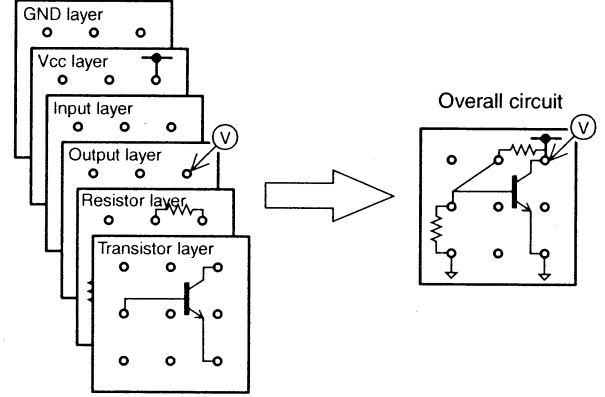


Figure 2: Overall circuit is generated from superimposed layers representing each type of element.

### 3 Design Procedure

In order to search for proper circuits that satisfy the specifications, we use a GA. However, there are some different points between our method and the simple GA. We use cells placed on the grids as genes instead of binary string. Each cell has a real number. The input to the automatic design system is specifications and constraints on available devices (c.f. Fig. 3).

The design procedure proceeds as follows. We first translate the specifications to the fitness function of the GA. Variables of fitness function are SPICE outputs, e.g. DC operating points, frequency responses, voltages of transient analysis, etc. The restriction of element types determines the number of layers. Secondly, the GA is run producing circuits having characteristics close to the given specification. Thirdly, generated circuits having the best fitness are considered as the solution. The solution is then simplified topologically to remove the “junk circuits”. The generated circuit can have same or near fitness when some sub-circuits or elements are removed. We call these “junk circuits”. They are mainly generated by the effect of hitchhiking.

The genetic operations consist of crossover and mutation. Crossover exchanges random selected rectangular area between two individuals. The area includes all layers that represent the elements. Mutation occurs with probability “Mutation Rate”. If the mutation occurs, it adds a random value selected from the interval  $[-2, 2]$  to the cell value.

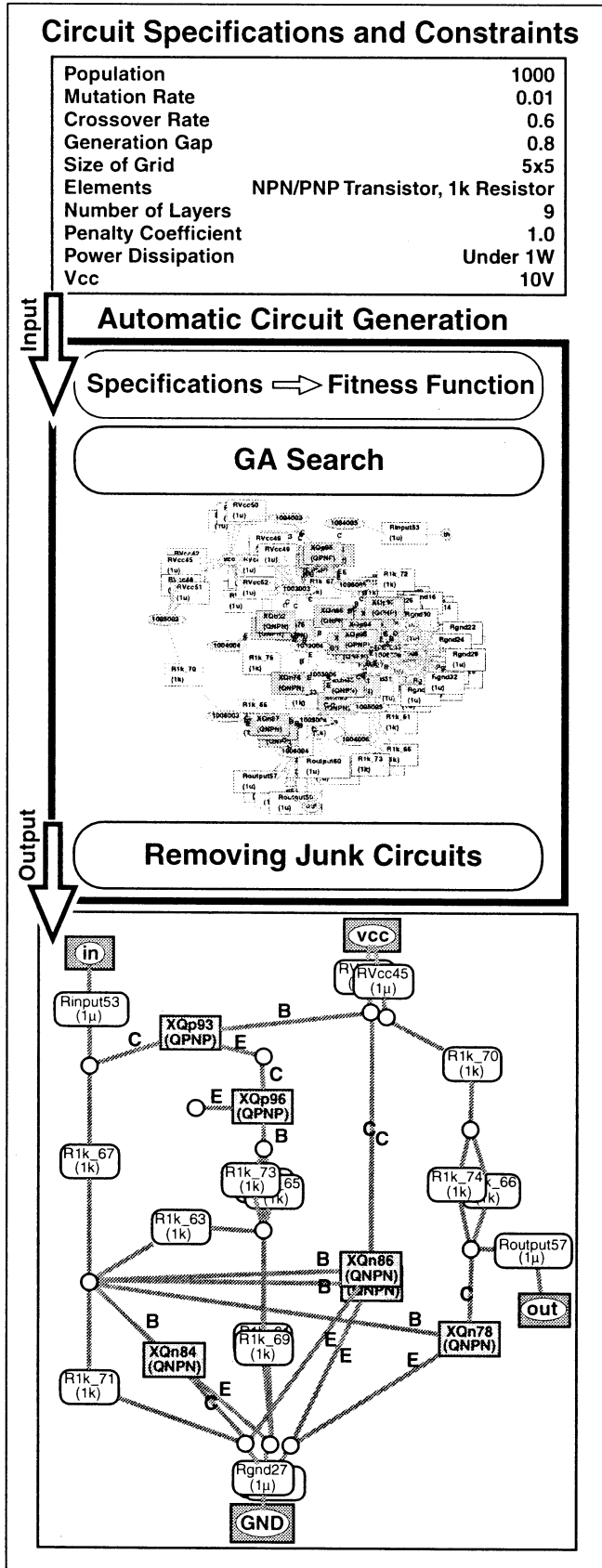


Figure 3: Design procedure.

## 4 Design Example

We selected a linear inverter as the objective of this experiment. A linear inverter is a circuit satisfying

$$V_{\text{Output}} = V_{\text{CC}} - V_{\text{Input}} \quad (2)$$

Other specifications are described in Fig. 3. The device model of transistors is based on SPICE default model; A diode is connected between base and emitter to simulate base emitter breakdown effect of the transistors. The added diode has device parameters:  $BV = 5V$  (breakdown voltage),  $IS = 1 \times 10^{-17}A$  (saturation current). Parameter  $IS$  is set to a tenth of the  $IS$  value of the transistor to have less effect on normal behavior.

### 4.1 Fitness Definition

Input voltage source of SPICE simulator supplies 0V to 10V by 0.5V to the circuit. Three quantities: output voltage, current flow of input voltage source and power supply are measured by SPICE. Fitness function is calculated as follows:

$$F = f_{\text{avg}} f_{\text{max}} f_{\text{VD}} f_{\text{RD1}} \frac{1}{f_{\text{RD2}} + 1} f_{\text{RA}} f_{\text{IVcc}} f_{\text{IVin}} \quad (3)$$

where,

$f_{\text{avg}}$ : Average of the error voltages that represent differences from ideal output.

$f_{\text{max}}$ : Maximum voltage error.

$f_{\text{VD}}$ : Deviation of measured output voltages.

$f_{\text{RD1}}$ : Vertical deviation of  $V_{\text{in}}-V_{\text{out}}$  curve after rotated -45 degrees around the point (5V,5V).

$f_{\text{RD2}}$ : Horizontal deviation of  $V_{\text{in}}-V_{\text{out}}$  curve after rotated -45 degrees around the point (5V,5V).

$f_{\text{RA}}$ : Horizontal average of  $V_{\text{in}}-V_{\text{out}}$  curve after rotated -45 degrees around the point (5V,5V).

$f_{\text{IVcc}} = \exp(-I(V_{\text{cc}}))$ .

$f_{\text{IVin}} = \exp(-I(V_{\text{in}}))$ .

### 4.2 Simulation Environment

The proposed evolutionary design method evaluates a large number of circuits. Since evaluation of the individuals in each generation can be done in parallel, the evaluation task is distributed to several host computers. We use common UNIX systems that are connected through TCP/IP. The performance varies according to the complexity of the circuits, but we achieved over 125 circuits/second in a system consisting of 4 Pentium II 350MHz machines and 4 Pentium 200MHz machines.

### 4.3 Results

The best DC transfer characteristics of different generations of a run with parameters given in Fig. 3 are shown in Fig. 4. The final solution is shown in Fig. 3. The circuit described in Fig. 3 is simplified by removing junk circuits. As a result of junk removal, the circuit elements are reduced from 36 to 17.

The generated circuit contains reverse connections of transistors that are inappropriate in many applications.

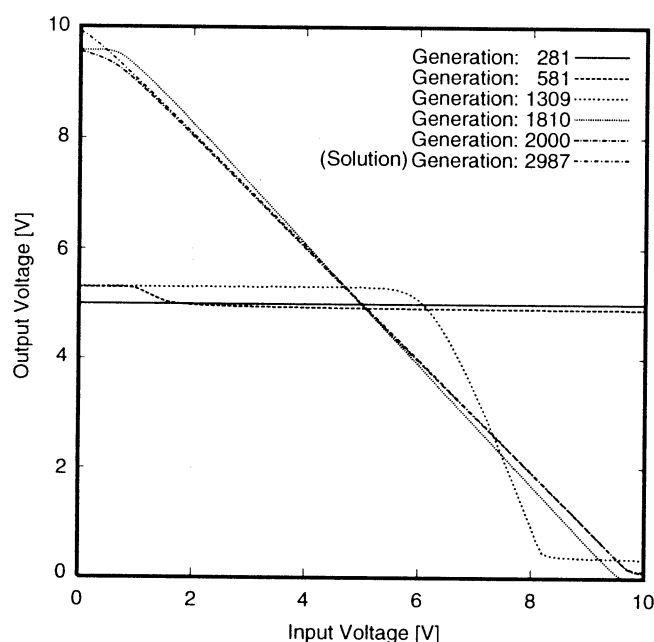


Figure 4: DC input-output characteristics of various generations.

### 5 Conclusions

We proposed a method for analog design automation method based on GA. A circuit representation scheme called PGA coding was also devised. PGA coding allows circuits with flexible topology. The ability of this evolutionary method to construct a desired inverter circuit from the given specifications was confirmed. Simulations revealed some problems of evolutionary-generated analog circuits. The first one is the existence of junk circuits, electrically excessive sub-circuits having no effect on electrical characteristics of the overall circuit. The junk circuits escape the evolutionary force of selection mainly due to the effect of hitchhiking and are propagated through generations to finally appear in the solution circuit. The second problem is the selection of transistor device model.

The GA takes advantage of holes in the model and makes inappropriate use of devices. The third problem is the selection of an effective fitness function. Circuit space is so large that simple error-related fitness functions could not lead the search to a solution.

Future research topics include comparison of the quality of the circuits obtained by our system to the results obtained through GP, and incorporating sizing information in the genotype.

### Acknowledgments

The authors wish to thank students K. Takada and Y. Maruyama of University of Electro-Communications for their technical assistance. Valuable comments by M. Dominguez Colonnier is also acknowledged.

### References

- [1] M. Toyama, M. Ito, "Optimization Method of small-signal amplifiers parameters using Genetic Algorithm," (in Japanese) *IEEJ Technical Meeting on Electronic Circuits*, ECT-98-8, pp. 33-38, 1998.
- [2] D. H. Horrocks, M. C. Spittle, "Component value selection for active filters using genetic algorithms," *Proc. of IEE Workshop on Natural Algorithms in Signal Processing*, Chelmsford UK, 14-16 Nov., Vol. 1, pp 13/1-12/6.
- [3] W. Kruiskamp and D. Leenaerts, "DARWIN: Analogue circuit synthesis based on genetic algorithms," *Int. J. Circ. Theor. Appl.*, vol. 23, pp. 285-296, 1996.
- [4] J. R. Koza, F. H. Bennett, III, D. Andre K. and F. Dunlap, "Automated synthesis of analog electrical circuits by means of genetic programming," *IEEE Trans. on Evolutionary Computation*, vol. 1, pp. 109-128, July 1997.
- [5] F. Gruau, "Cellular encoding of genetic neural networks," *Technical report 92-21*, Laboratoire de l'Informatique du Parallélisme, Ecole Normale Supérieure de Lyon, 1992.
- [6] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading MA USA, 1989.
- [7] T. Quarles, D. Pederson, R. Newton, A. Sangiovanni-Vincentelli, and Christopher Wayne, *SPICE 3 Version 3F5 User's Manual*, Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA, March 1994.

# Periodic Motion Generated After Chaos in a Model of Trading Agents

Mieko TANAKA-YAMAWAKI and Masayoshi TABUSE  
Department of Computer Science and Systems Engineering  
Faculty of Engineering, Miyazaki University  
mieko@cs.miyazaki-u.ac.jp

## Abstract

We have observed in our model of trading agents a phenomenon similar to the transient chaos. Unlike the cases in other well-known dynamical systems, the time series that we deal with are the average of the prices of entire agents in the system hence do not follow any deterministic law by itself, though the price of each agent obeys a deterministic law. Depending on the parameters of the model, the long-term series of the average prices undergo transitions from chaotic behavior to the motion of a single period in a self-organized manner. Once the system reaches this periodicity, it persists forever, like a limit cycle. When this occurs it is seen that the active agents move with groups of two or more agents, which can be interpreted as a natural formation of cooperation.

**Key Words:** economic system, trading agents, transient chaos, random walk, Lévy distribution

## 1. Introduction

The boundary of chaos and periodicity ('edge of chaos') attracts much attention as a clue to understand complex systems such as living systems characterized by the 'emergent' property. In this line of thoughts, we have been studying a closed system of trading agents which generates chaotic time series.

We have seen in this model various interesting properties of nonlinear systems. Particularly, the long-term time series of average price generated by the computer simulation of this model exhibit three stages:

First phase: In this phase the whole system moves coherently. The corresponding power spectrum has a large peak at low frequency range and has  $f^{-2}$  behavior at high frequency range, and the largest Lyapunov exponent is positive and tiny, showing that this system is close to the edge of chaos.

Second phase: The coherent motion suddenly collapses to a random motion. In this phase the largest Lyapunov exponent is still positive and tiny and the power spectrum is still  $f^{-2}$ . Only the large peak at low frequency range, that works to form an envelope to cause the coherent motion, is missing.

Third phase: In this phase the time series become entangled to completely periodic motions. This phenomena occurs after many trading agents retire from the trade by losing their assets (oligopoly).

In the present paper we concentrate on the problem of the periodic motions that appear at the third stage of the long-term time series after weak chaos. Most notable fact is that the agents are factored into several groups of size 2 and the members of the same group take the same action each other, which is regarded as a natural formation of cooperation.

## 2. A Model of Trading Agents

We shall briefly sketch the outline of our model of

trading agents[1]. Our motivation is to deal with the dynamical feature of the social activity without relying on the conventional approach of setting a few equations of motion with a small number of global parameters. Particularly we are interested in modeling economics-related phenomena such as trade cycles directly using the microscopic variables so that we can explicitly trace the change of those variables by means of computer simulation, which is essential for the study of the non-equilibrium system.

The trading rules are set to have a positive feedback property as follows: Each agent (numbered by  $k$ ) increases the price  $P_k(t)$  of goods and its budget  $B_k(t)$  for investment, if this agent successfully sold the goods at  $t$ :

$$P_k(t+1) = P_k(t) + b \quad (1)$$

$$B_k(t+1) = B_k(t) + b \quad (2)$$

If, on the other hand, no goods were sold at time  $t$ , then it lowers the price for the next trade and shrinks the budget for investment.

$$P_k(t+1) = P_k(t) / (1 + a) \quad (3)$$

$$B_k(t+1) = B_k(t) / (1 + a) \quad (4)$$

The price and the budget are unchanged if no sales are made due to the lack of goods to sell at time  $t$ :

$$P_k(t+1) = P_k(t) \quad (5)$$

$$B_k(t+1) = B_k(t) \quad (6)$$

The model has three positive constant parameters  $\{a, b, L\}$ , where  $L$  is the time delay after the purchase to the time to sell,  $(1 + a)$  is the suppression factor of the price and the budget, and  $b$  is the increment of these two variables.

The time delay  $L$  plays an essential role in this model to stabilize the synchronized motion. As we increase  $L$ , beginning from its minimum value one, the system maintains its stability longer. This effect may be interpreted as a proof for an existence of controllable trade cycle caused by a long inventory period [2]. For a small system made of, e.g., ten agents, it is quite difficult to find the initial condition to maintain the system stable for  $L=1$ . Patterns appearing in this unstable region, however, provides us many examples of interesting chaotic behavior. We shall concentrate in this paper on analyzing these chaotic time series in the transient regime of chaos.

To be specific, we look at the time series of the

average price for the case of one hundred agents at  $L=1$ . A typical pattern is shown in Figure 1 and Figure 2 for  $a=0.06$ ,  $b=0.03$  for the case without the periodic phase.

### 3. Self organization toward the periodicity

For some values of the parameters, the average price automatically turns to a periodic motion. A typical example is the case of shown in Fig.3 for  $a=0.06$ ,  $b=0.08$ . Both have the coherent motion of large amplitude in the beginning, then suddenly change to a random pattern of much smaller amplitude. The former case(Fig.1) maintains this second phase forever(we have checked up to one million trading times), while the latter case(Fig.3) further shifts itself to a simple periodic motion by a few remaining agents after many fellows have dropped out of the trading activity, i.e., when the system is in the state of extreme oligopoly. This process is shown in Fig. 4-6.

The transition from the first phase to the second phase occurs almost suddenly as can be seen in Figure 2 and Figure 4. We discussed the use of ‘n-down curve’ to understand the underlying mechanism of this rapid transition of the phases in the previous meeting[3]. Note that these two phases are both weakly chaotic in the sense that the first Lyapunov exponents are small positive values. .

### 4. Stability of the periodic state

Now we investigate how stable the periodic trading system is. To see this we have ‘kicked’ each agent’s price by a small positive amount (but not larger than 0.04 for each agent) at  $t=94,000$ . As shown in Fig. 7, the deviation recovers within 100 times of trades to the original periodic motion. Fig. 8 shows the case for the negative kick by 0.04 for each agent. Thus we conclude that this periodic system is highly stable against external disturbance once it is formed.

### 5. Relation to the statistical property

Recently, there has been a renewed interest in the statistical property of the economic time series.

Mantegna and Stanley[4] have shown that the price fluctuation of Standard and Poor 500 Index follows Lévy's stable distribution:

$$f(x, \alpha, c) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \exp(ikx - c|k|^\alpha) dk$$

of  $\alpha=1.4$ , instead of Gauss distribution

$$f(x, \alpha = 2, c) = \frac{1}{2\sqrt{c\pi}} \exp\left(-\frac{x^2}{4c}\right)$$

which corresponds to a special case of  $\alpha=2$  in the above distribution.

The same analysis for the minutely data of Tokyo Stock Market Index (TOPIX) is reported to be the same distribution with the same parameter value [5]. We tried to see whether our model generates the same price change as the real stock market indices and analyzed the above mentioned two cases:  $a=0.06$ ,  $b=0.03$  as an example of having no third phase, and  $a=0.06$ ,  $b=0.08$  as an example of the case attracted to a periodic motion.

The result is a striking one. The case of no third phase shows Gauss distribution and the case with the third phase exhibits the characteristic behavior of Lévy's stable distribution with  $\alpha=1.9$ . We shall report this result in more details elsewhere[6].

## 6. Conclusion

We have reported the observation that time series turn from the chaotic behavior to the periodic pattern as the agents drop out of the trade and the remaining active agents automatically form groups inside a system. It is found that this periodic system thus formed is very stable under external disturbances. Also the statistical property of the changes is observed to follow Lévy's stable distribution with  $\alpha=1.9$  for the class of time series having the third phase (periodic phase), while the class of time series without a periodic phase seems more like Gauss distribution.

## Acknowledgments

This work is supported in part by a General Research Grant (C2:10680361) by the Ministry of Education, Science and Culture, Japan.

## References

- [1] Mieko Tanaka-Yamawaki and Masayoshi Tabuse, Proceedings of Iizuka'98 (Oct. 1998) pp.884-887; M.Tanaka-Yamawaki, K.Hasebe and L.C.Jain, Proceedings of EDT2000 (IEEE Computer Soc.,1995) pp.571-577.
- [2] Immanuel Wallerstein ed. "Long Waves", Fernaud Braudel Center and The Research Foundation of the State University of New York, (Essays Selected from Review).
- [3] M.Tabuse and M.Tanaka-Yamawaki, Proceedings of AROB 3<sup>rd</sup> '98 (1998) pp.482-485.
- [4] R. N. Mantegna and H. Eugene Stanley, Nature, vol. 376 (1995) pp.46-49.
- [5] K. Yanagawa, Master Thesis: Nihon University, March, 1998 (in Japanese).
- [6] Mieko Tanaka-Yamawaki and Masayoshi Tabuse, in preparation for an IEEE meeting.

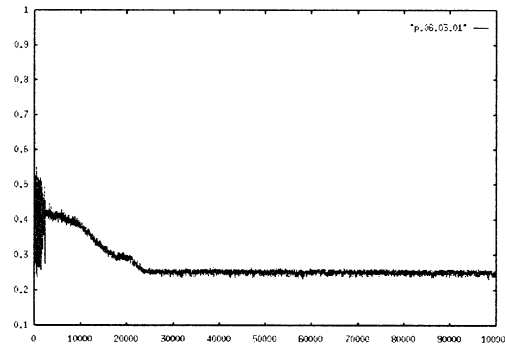


Figure 1 Time series of the average price for the first data set ( $a=0.06$ ,  $b=0.03$ ,  $L=1$ ) for 100 agents.

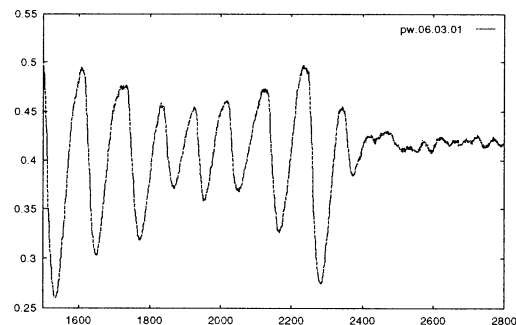


Figure 2 Enlarged time series in Figure 1 near the transition from the first phase to the second phase.

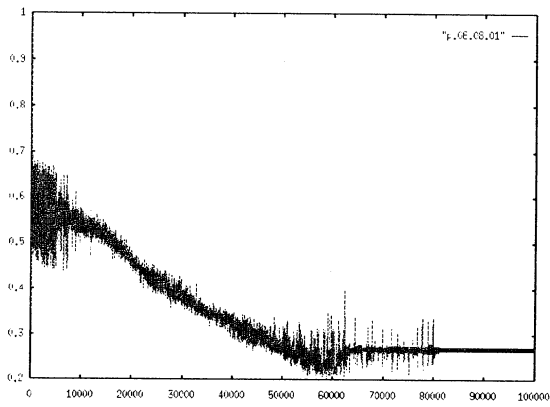


Figure 3 Time series of the average price for the parameter values of  $a=0.06$ ,  $b=0.08$ ,  $L=1$ , for 100 agents.

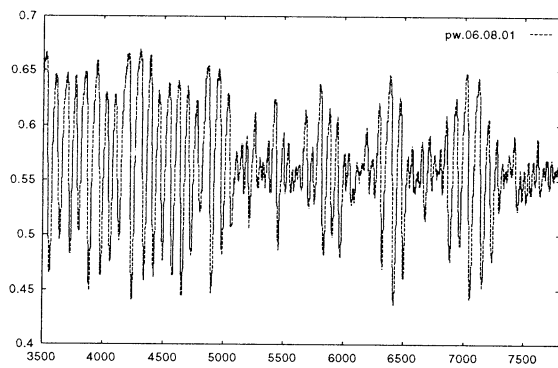


Figure 4 Enlarged time series in Figure 3 near the transition from the first phase to the second phase. Temporal returns to the first phase can be observed at the beginning of the second phase, similar to the on-off intermittency.

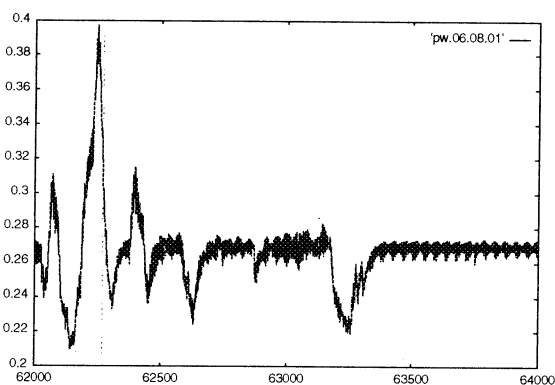


Fig. 5 The average price undergoes a transition from chaotic pattern to quasi-periodic pattern for  $a=0.06$ ,  $b=0.08$ ,  $L=1$ , for 100 agents.

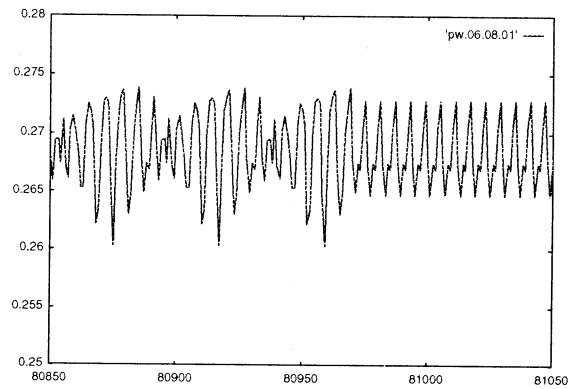


Fig. 6 The average price undergoing a transition from quasi-periodic pattern to periodic pattern for  $a=0.06$ ,  $b=0.08$ ,  $L=1$ , for 100 agents.

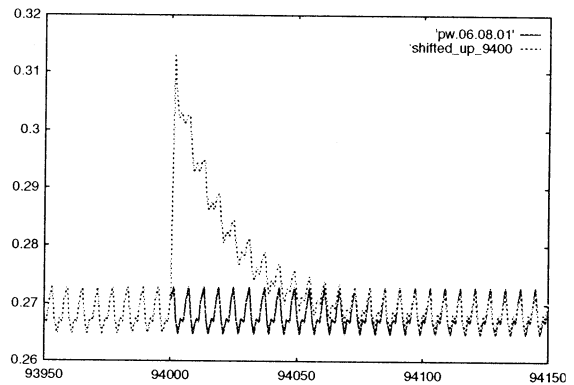


Fig. 7 Stability of the periodic pattern is confirmed by observing that shifting the price of each agent by 0.04 at  $t=94,000$  is quickly tamed within 100 turns of trades.

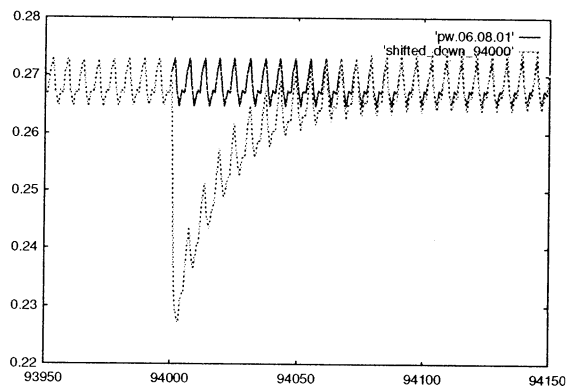


Fig. 8 Similar to the above figure, shifting of the price of each agent by  $-0.04$  at  $t=94,000$  is also quickly tamed within 100 turns of trades.



# A Study of Chaos Associative Memory

Masahiro Nakagawa

Nagaoka University of Technology,  
Kamitomioka 1603-1, Nagaoka, Niigata 940-2188, Japan  
masanaka@vos.nagaokaut.ac.jp

**Abstract** - In this paper we shall propose a novel chaos neural network model applied to the autoassociative memory. The present artificial neuron model is properly characterized in terms of a time-dependent periodic activation function to involve a chaotic dynamics as well as the energy steepest descent strategy.

## I. INTRODUCTION

During the past quarter century, a number of autoassociative memory models have been extensively investigated on the basis of the autocorrelation dynamics. Since the foundations of the retrieval models by Anderson<sup>1</sup>, Kohonen<sup>2</sup>, and Nakano<sup>3</sup>, some works related to such an autoassociation model of the interconnected neurons through an autocorrelation matrix were theoretically analyzed by Amari<sup>4</sup>, Amit *et al.*<sup>5</sup> and Gardner.<sup>6</sup> So far it has been well appreciated that the storage capacity of the autocorrelation model, or the number of pattern vectors,  $L$ , to be completely associated vs the number of neurons, which is called the relative storage capacity or storage capacity for short, is estimated as  $\alpha_c = L/N \sim 0.14$  at most for the autocorrelation learning model with the activation function as the signum one ( $\text{sgn}(x)$  for the abbreviation), where  $N$  is the number of neurons.<sup>7,8</sup>

From an aspect concerned with chaotic dynamics, Tsuda reported some works concerned with dynamic retrieval model, dynamic linking of associative memories and explored the significance of chaos dynamics.<sup>9,10</sup> Nara *et al.* also argued the memory search model with a chaos control.<sup>11</sup> So far some applications of the chaotic neural networks have been reported by Aihara *et al.*,<sup>12</sup> Nakamura and Nakagawa.<sup>13</sup> In practice, however, as was confirmed by Kasahara and Nakagawa,<sup>14</sup> the chaotic dynamic association has been found to encounter the problem such that the complete association of the embedded patterns becomes inevitably troublesome if the loading rate, i.e.  $\alpha_c$  is increased beyond  $\sim 0.2$  even though the orthogonal learning model with the generalised inverse matrix is utilised. Also the chaotic dynamics with a monotonous activation function was applied to such a combinatorial

optimization problem as the Traveling Salesman Problem (TSP).<sup>14</sup> They clarified that a parameter controlled chaos dynamics has a capability to realise a more efficient search of an optimal solution in TSP beyond the earlier work with fixed parameters reported by Nozawa.<sup>15</sup>

In contrast to the above-mentioned models with monotonous activation functions, the neuro-dynamics with a nonmonotonous mapping was recently proposed by Morita,<sup>16</sup> Yanai and Amari,<sup>17</sup> Shiino and Fukai.<sup>18</sup> They reported that the nonmonotonous mapping in a neuro-dynamics possess a certain advantage of the storage capacity,  $\alpha_c \sim 0.27-0.42$ , superior than the conventional association models with such a monotonous mapping as the signum or sigmoidal function.

Combining the above-mentioned two strategies, i.e. the chaos dynamics and the nonmonotonous dynamics, the present author proposed a novel neuron model with a periodic activation function to construct an association model with chaotic dynamics as the discrete time and orthogonal learning model.<sup>19-22</sup> Recently such a chaotic dynamics was involved in the synergetic neural network<sup>23</sup> to construct a chaos synergetic neural network model which involves a competition dynamics between overlaps.<sup>24,25</sup> In the single layer structure association model, however, the memory capacity has not been dramatically improved even by the chaotic neural networks.<sup>19-22,26</sup> Further an extension of the external chaos control models was put forward to elucidate an efficiency of an autonomous control based on the energy functional.<sup>27,28</sup>

As described above, the periodic activation function has been shown to be eventually efficient in the orthogonal learning model with the generalised inverse matrix. To date, however, there has been no report concerned with the autocorrelation dynamics in such a periodic chaos neuron model.<sup>19-22,26-28</sup>

In the present paper, let us propose a chaos associative memory (CAM) model with the autocorrelation dynamics instead of the previously concerned orthogonal models.<sup>19-22,26-28</sup>

## II. THEORY

First of all let us define some dynamic rules to construct a chaotic neural network below. For this

purpose we shall define first the internal state and the corresponding output of the  $i$ th neuron as  $\sigma_i$  and  $s_i$ , respectively, which have to be related to each other in terms of the following sinusoidal mapping,

$$s_i = f_i(\sigma_i) = \sin\left(\frac{\pi}{2} \frac{\sigma_i}{\tau}\right). \quad (1)$$

The energy function of the dynamical system with  $N$  neurons may be defined by

$$E = E_w + E_c, \quad (2)$$

where an objective function,  $E_w$ , and a coupling energy function,  $E_c$  are defined by

$$E_w = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij} s_i s_j, \quad (3)$$

and

$$E_c = \sum_{i=1}^N \lambda_i \int ds_i \sigma_i + \frac{1}{4} \sum_{i=1}^N \gamma_i (1 - s_i^2)^2, \quad (4)$$

respectively; here  $w_{ij}$  ( $1 \leq i, j \leq N$ ) are the autocorrelation memory matrix components corresponding to the interconnection strengths between the  $i$ th and the  $j$ th neurons and assumed to be symmetric, i.e.  $w_{ij} = w_{ji}$ ,  $\lambda_i$  ( $1 \leq i \leq N$ ) are the coupling constants between  $\sigma_i$  (internal state) and  $s_i$  (output), and  $\gamma_i$  ( $1 \leq i \leq N$ ) are the Lagrange multiplier of the  $i$ th neuron corresponding to the constraints  $|s_i| = 1$  ( $1 \leq i \leq N$ ) at a retrieval point. As in the conventional autocorrelation learning model with off-diagonal components,  $w_{ij}$  can be simply defined by

$$w_{ij} = \frac{1}{N} \sum_{r=1}^L (e^{(r)}_i e^{(r)}_j - \delta_{ij}), \quad (5)$$

where  $L$  is the number of the embedded pattern vectors and  $e^{(r)}_i$  ( $= \pm 1$ ) is the  $i$ th component for the  $r$ th embedded pattern, and all the embedded vectors are assumed to be linearly independent each other.

Then the innerproducts  $\mu^{(r)}$  ( $r=1, 2, \dots, L$ ), which are regarded as the pattern matching rates, are to be defined as follows,

$$\mu^{(r)} = \frac{1}{N} \sum_{i=1}^N e^{(r)}_i s_i \quad (1 \leq r \leq L). \quad (6)$$

Now let us define the dynamics of the present system below. The time-dependent Ginzburg-Landau (TDGL) equation of the internal state  $\sigma_i$  may be given by

$$\frac{D\sigma_i}{Dt} = -\frac{\partial E}{\partial \sigma_i}, \quad (1 \leq i \leq N) \quad (7)$$

where the operator,  $D \cdot / Dt$ , may be regarded as the forward time difference operator (for the discrete-time

model) or the time differential operator (for the continuous-time model). Substituting eq.(2) into eq.(7), one readily has

$$\frac{D\sigma_i(t)}{Dt} = \lambda_i \sigma_i(t) + \sum_{j=1}^N w_{ij} s_j(t) + \gamma_i (1 - s_i(t)^2) s_i(t). \quad (8)$$

In the above dynamic equations, the coupling constant  $\lambda_i$  is concerned with the relaxation time of the  $i$ th neuron. If one resorts on the discrete-time model under consideration, the difference operator can be replaced as follows,

$$\frac{D\sigma_i(t)}{Dt} = \frac{\sigma_i(t+h) - \sigma_i(t)}{h}, \quad (9)$$

where  $h$  is the time division interval for the  $t$ -axis. For a continuous-time model one may assume  $h \rightarrow 0$ , where the forward operator and the backward one will be equivalent to each other. Making use of eq.(9), our dynamic equation (8) reads

$$\begin{aligned} \sigma_i(t+h) &= (1 - \lambda_i h) \sigma_i(t) \\ &+ h \sum_{j=1}^N w_{ij} s_j(t) + h \gamma_i (1 - s_i(t)^2) s_i(t) \end{aligned} \quad (10)$$

$$= (1 - \lambda_i h) \sigma_i(t) + h \sigma^*_{i}(t)$$

where  $\sigma^*_{i}(t)$  is defined by

$$\sigma^*_{i}(t) = \sum_{j=1}^N w_{ij} s_j(t) + \gamma_i (1 - s_i(t)^2) s_i(t). \quad (11)$$

It should be borne in mind here that  $s_i(t)$  and  $\sigma_i(t)$  have to be related each other in terms of eq.(1), and that  $\tau$  is assumed to be controlled towards 1 at a retrieval point, or a fixed point corresponding to a basin in the  $N$ -dimensional phase space spanned by  $s_i$  ( $1 \leq i \leq N$ ).

According to the above-noted idea, in this work, we simply assume the following linear dynamics for  $\tau(t)$ .

$$\frac{D\tau}{Dt} = \kappa (1 - \tau), \quad (12)$$

where  $\kappa$  is a positive constant to drive the system from chaotic ( $\tau(t) \sim 0$ ) to nonchaotic ( $\tau(t) \sim 1$ ) state.

### III. RESULTS

We shall show a few examples of the dynamic behaviour of the present model in a chaotic associative mode with eqs.(10) and (11). Hereafter  $h$ ,  $\kappa$ ,  $\eta$ ,  $\lambda_i$ ,  $\gamma$ ,  $\tau_0$  and  $\epsilon$  are set to 0.5, 0.8, 0.03, 1, 1,  $10^{-4}$ , and 10, respectively, if not mentioned below. The embedded pattern vectors were randomly selected from  $2^N$  binary patterns, and put into

$$e^{(r)}_i = \text{sgn}(z^{(r)}_i) \quad (1 \leq r \leq L, 1 \leq i \leq N), \quad (13)$$

where  $z_i^{(r)}$  ( $1 \leq i \leq N$ ,  $1 \leq r \leq L$ ) are the pseudo-random numbers between -1 and +1 with zero-mean value.

In the present associative model, let us investigate the dynamic memory retrieval characteristics with eq.(11) in the autoassociation mode with a key input vector  $\theta_i$ . In analogous to the previous models, <sup>26,27</sup> eq.(1) may be extended as

$$s_i(t) = \sin\left(\frac{\pi}{2} \frac{\sigma_i(t) + \beta(\tau)\theta_i}{\tau(t)}\right), \quad (14)$$

where  $\theta_i$  is the initial input vector for the autoassociation with the Hamming distance  $H_d$ , which corresponds to the number of the components to be set into 0, from a target pattern  $e^{(s)}_i$  ( $1 \leq s \leq L$ ), and  $\beta(\tau)$  such that  $\beta(\tau) \rightarrow 0$  as  $\tau \rightarrow 0$  or  $\tau \rightarrow 1$  is assumed to be given by

$$\beta(\tau) = \tau(t)^\eta \cdot (1 - \tau(t))^\eta, \quad (15)$$

where  $\eta$  is a positive exponent. From eqs.(14) and (15), one may realise an addition of the key information  $\theta_i$  during the transient chaotic state between  $\tau \sim 0$  and  $\tau \sim 1$ . The directional cosine,  $\zeta^{(s)}$ , of the initial input vector  $\theta_i$  with respect to a target pattern  $e^{(s)}_i$  can be evaluated in terms of

$$\zeta^{(s)} = \frac{1}{N} \sum_{i=1}^N e^{(s)}_i \theta_i = 1 - \frac{H_d}{N}. \quad (16)$$

Now choosing the parameters  $N$ ,  $\gamma$  and  $\eta$  as 100, 1 and 0.03, respectively, the memory capacities are derived as in Figs.1(a) and (b) for  $H_d/N = 0.01$  and 0.3, respectively. Thus we may confirm the memory retrieval up to  $\alpha_c \sim 0.7$  which is remarkably larger than the corresponding value derived in the conventional models with the monotonous activation function. To see this fact, we present the results for  $H_d/N = 0.01$ , and 0.3 with the monotonous associatron model in Figs.2(a) and (b), respectively, in which eqs.(1), (10) and (11) were replaced by

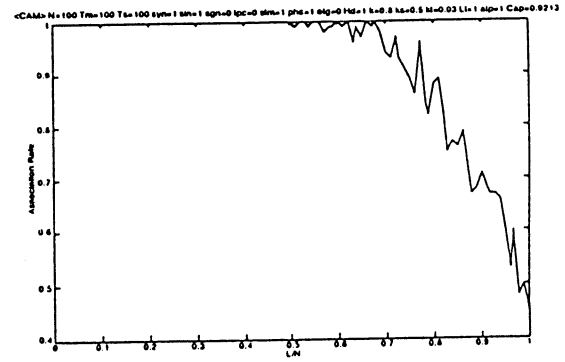
$$s_i(t) = f(\sigma_i(t)) = \text{sgn}(\sigma_i(t)), \quad (17)$$

$$\sigma_i(t+1) = \sum_{j=1}^N w_{ij} s_j(t), \quad (18)$$

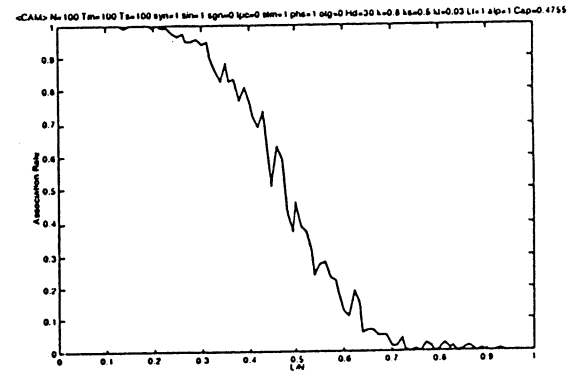
where  $w_{ij}$  is defined again in terms of eq.(6), together with the following initial condition

$$s_i(0) = \theta_i. \quad (19)$$

From comparison between Fig.1 (sinusoidal) and Fig.2 (signum), one may confirm the advantage of the present model with a periodic activation function whose total memory capacity,  $M_c$ , defined as the total area of the characteristic curve of success rate vs loading rate, is evaluated as  $M_c \sim 0.92$  (Fig.1(a)) as  $H_d/N \rightarrow 0$  beyond  $M_c \sim 0.18$  (Fig.2(a)) for the signum as  $H_d/N \rightarrow 0$ .

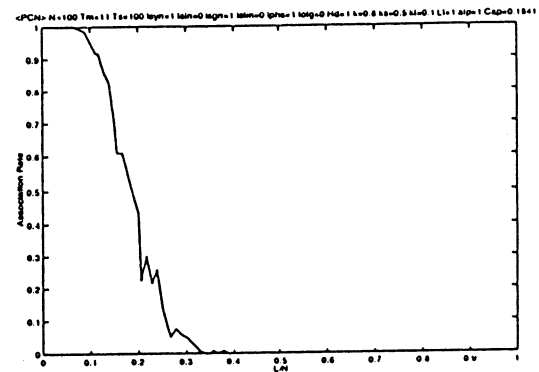


(a)  $H_d/N = 0.01$

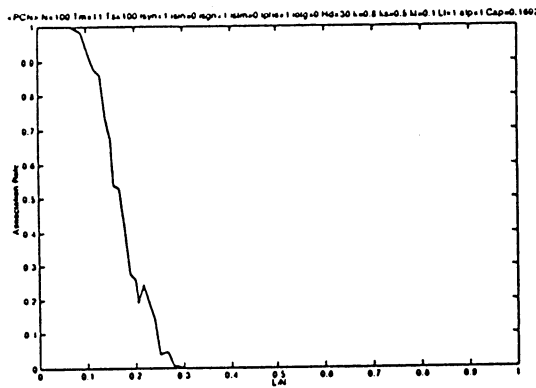


(b)  $H_d/N = 0.3$

Fig.1 The autoassociation characteristics with  $s_i(t) = \sin\left(\frac{\pi}{2} \frac{\sigma_i(t) + \beta(\tau)\theta_i}{\tau(t)}\right)$ , where  $\theta_i$  is an input vector with the Hamming distance  $H_d$  from a target pattern  $e^{(s)}_i$ . Here  $N=100$ . Here  $N$ ,  $\gamma$  and  $\eta$  are set to 100, 1 and 0.03, respectively.



(a)  $H_d/N = 0.01$



(b)  $H_d/N=0.3$

Fig.2 The autoassociation characteristics with  $s_i(t) = \text{sgn}(\sigma_i(t))$ , where  $\theta_i$  is an input vector with the Hamming distance  $H_d$  from a target pattern  $e^{(s)}_i$ . Here  $N=100$ . Here  $N$ ,  $\gamma$  and  $\eta$  are set to 100, 1 and 0.03, respectively.

#### IV. CONCLUDING REMARKS

In this paper we have constructed a simple chaotic associative memory, CAM model with autocorrelation dynamics characterised by the periodic activation function,<sup>29</sup> which maps the internal state  $\sigma_i$  to the output  $s_i$ , for each neuron with the periodicity  $\tau$  as defined by eq.(1). While such a periodic mapping does no longer assure the monotonously decreasing energy as substantially involved in the monotonously increasing activation function, it may prevent the system from unfavourable trappings corresponding to spurious states as was previously noted.<sup>19-22</sup> In practice the dynamic memory retrieval characteristics of such a periodic chaos neural network with periodicity control has been found to be considerably improved in comparison with the conventional chaos neural networks with such a monotonous mapping as a sigmoidal function<sup>12,13</sup> Figure 2(a) shows that the association can be realised up to the loading rate  $\alpha_c \sim 0.7$  with  $H_d/N \rightarrow 0$  beyond the previous finding derived from the partial reverse dynamics with the discrete time proposed by Morita et al<sup>16,30</sup>, in which  $\alpha_c \sim 0.27$  at most even for  $H_d/N \rightarrow 0$ . In addition the total memory capacity,  $M_c$ , of the presently proposed autoassociation model, which is defined as the total area of the memory characteristic curves as depicted in Fig.1(a), was found to be  $M_c \sim 0.92$  and be promoted up to  $\sim 5$  times larger than the monotonous association model, i.e. the associatron as defined in terms of eqs.(17) and (18), in which  $M_c \sim 0.18$  as shown in Fig.2(a). From these findings, it may be concluded that the analogue periodic mapping with a chaotic dynamics has a certain advantage

beyond the previously proposed monotonous mapping<sup>12,13</sup> as well as the previously proposed nonmonotonous one.<sup>16,30</sup>

As a future problem, it seems to be fascinating to extend the present model with continuous time instead of the discrete one, and also to investigate the autonomous control model with the autocorrelation dynamics.<sup>27,28</sup> As a further progress, it is considered to be worthwhile to construct the multilayered model as the error back-propagation neural network,<sup>31</sup> and to extend the present model to the case of the analogue pattern vectors, whose components may range over -1 and +1, instead of the presently concerned binary vectors together with a synergetic dynamics,<sup>23,24</sup> Nakagawa,<sup>32</sup> Kitahara and Nakagawa.<sup>33</sup>

#### References

1. Anderson J. A. (1972): Math. Biosci. 14, 197.
2. Kohonen T. (1972): Correlation matrix memories, IEEE Trans. C-21, 353.
3. Nakano K. (1972): IEEE Trans. SMC-2, 380.
4. Amari S.: (1977) Biol. Cybern. 26, 175.
5. Amit D. J., Gutfreund H., and Sompolinsky H. (1985): Phys. Rev. Lett. 55, 1530.
6. Gardner E. (1986): J. Phys. A 19 L1047.
7. McEliece R. J., Posner R. J., Rodemich E. R. and Venkatesh S. S. (1987): IEEE Trans. on Information Theory, IT-33, 461.
8. Amari S. and Maginu K. (1988): Neural Networks 1, 63.
9. Tsuda I. (1991): Neurocomputers and Attention 1, (eds. A. V. Holden and V. I. Kryukov, Manchester Univ. Press, 405.
10. Tsuda I. (1992): Neural Networks, 5, 313.
11. Nara S., Davis P. and Totsuji H. (1993): Neural Networks, 6, 963.
12. Aihara K., Takabe T. and Toyoda M. (1990): Phys. Lett. A 144, 333.
13. Nakamura K. and Nakagawa M. (1993): J. Phys. Soc. of Jpn. 62, 2942.
14. Kasahara T. and Nakagawa M. (1995): Electronics and Communications in Japan Part III-Fundamentals, 78, 1.
15. Nozawa H. (1992): Chaos, 2, 377.
16. Morita M. (1993): Neural Networks, 6, 115.
17. Yanai Hiro-F. and Amari S. (1993): Proc. of ICNN'93, San Francisco, 1385.
18. Shiino M. and Fukai T. (1993): Phys. Rev. E 48, 867.
19. Nakagawa M. (1994): Proc. of ICONIP'94, Seoul, 1, 609.
20. Nakagawa M. (1995): Proc. of ICDC'94, Tokyo, 2, 603.
21. Nakagawa M. (1995): J. Phys. Soc. Jpn. 64, 1023.
22. Kasahara T. and Nakagawa M. (1995): J. Phys. Soc. Jpn. 64, 4964.
23. Nakagawa M. (1995): IEICE Trans. on Fundamentals E78-A, 412.
24. Nakagawa M. (1995): J. Phys. Soc. Jpn. 64, 3112.
25. Nakagawa M. (1995): Proc. of ICNN'95, Australia, 3028.
26. Nakagawa M. (1997): J. Phys. Soc. Jpn. 66, 263.
27. Nakagawa M.: J. Phys. Soc. Jpn. 65 (1996) 1859.
28. Nakagawa M. (1996): Proc. of ICNN'96, 2, 862.
29. Tanaka T. and Nakagawa M.: IEICE Trans. on Fundamentals J79-A, (in Japanese) 1826.
30. Morita M., Yoshizawa S., and Nakano K. (1990): Proc. of INNC'90, 2, 868-871.
31. Maeda H. and Nakagawa M. (1996): Proc. of ICONIP'96, Hong Kong, 1, 567.
32. Nakagawa M. (1997): IEICE Trans. on Fundamentals E80-A, 881.
33. Kitahara J. and Nakagawa M. (1997): IEICE Trans. on Fundamentals J80-A, (in Japanese) 949.

# A Chaotic Synthesis Model of Vowels

Hiroyuki Koga

Masahiro Nakagawa<sup>†</sup>

Department of Electrical Engineering, Faculty of Engineering, Nagaoka University of Technology,  
Kamitomioka 1603-1, Nagaoka, Niigata 940-2188, JAPAN

Tel : 0258-46-6000 (Ext.5476) E-mail : koga@pelican.nagaokaut.ac.jp masanaka@vos.nagaokaut.ac.jp<sup>†</sup>

Key words : Vocal Sounds , Chaos , Lyapunov Dimension

## Abstract

In the field of synthesis of vocal sounds, various models composed of glottal sound source, vocal tract, and radiation into space, have been suggested. These models however do not consider how to involve the chaos property into them which are not chaotic. In the present work, we proposed an oscillation model of voice synthesizer with chaotic behavior. This model utilized an exponential spring and damping in the self-oscillating model of the vocal cords. The present results show that the synthesized voices involve the chaotic property by means of the Lyapunov analysis.

## 1 Introduction

The chaotic concept<sup>1</sup> has 3 features generally. There are the orbital instability, the long-term unpredictability and the self-similarity. The research to express these features become active. The orbital instability and the long-term unpredictability which are sensitively dependent on initial conditions can be evaluated by the Lyapunov exponent and the KS entropy. The self-similarity structure of the attractor, can be also quantified by the non-integer fractal dimension<sup>2</sup>.

Since the vocal sound uttered by human has the fluctuation and the self-similarity, the fractal properties in the vocal sound has been found<sup>3</sup>. Their properties can do quantification by the Lyapunov spectrum analysis<sup>4</sup>. Since the maximum Lyapunov exponent becomes positive value for the vocal sounds, we can find out that vocal sounds contain the chaotic property in themselves.

Various field about vocal sounds, the modeling of the utterance system, the feature analysis, the recognition, and the vocal information system have been investigated. Various vocal synthesis model<sup>5-7</sup> to produce the artificial vocal sounds have been also studied, especially the model to produce the chaotic fluctuation, as the single mass model oscillating non-linearly is suggested by Yamaguchi, et al.<sup>8</sup>

Therefore we developed this model further, and

constructed the synthesis model of vocal sounds involving the chaotic behavior, also analyzed the Lyapunov spectrum of the synthesis vocal sounds.

## 2 The Vocal-Cord Model

We use the two-mass model by Ishizaka, et al.,<sup>7</sup> whose behavior is physiologically. A schematic diagram of this model is shown in Fig. 1. The vocal cords are assumed to be bilaterally symmetric, and we discuss the properties of only one side cord.

In the present model, two-cords consist of a simple mechanical oscillator having masses, are denoted as  $m_1, m_2$ , the lateral displacements,  $x_1, x_2$ , the restoring forces of nonlinear springs,  $\psi_1, \psi_2$ , the nonlinear viscous losses,  $\xi_1, \xi_2$ . Also the masses are coupled by a linear spring indicated as  $s_c$ . In addition, we denote subglottal pressure as  $P_s$ , the effective length of the cords  $l_g$ , the thickness of masses  $d_1, d_2$ , the cross-sectional areas of the glottal  $A_{g1}, A_{g2}$ , and the average volume velocity across the glottal area  $U_g$ , as shown in Fig. 1.

The widths between opposite cords  $h_1, h_2$  are described by  $h_i = x_i + A_{g0}/2l_g = x_i - x_c$ , ( $i=1,2$ ), respectively, where  $x_c$  is the displacement at the cords closure, and  $A_{g0}$  is the "phonation neutral" area. Therefore  $A_{g1}$  and  $A_{g2}$  are expressed by  $A_g = 2l_g h_i$ , ( $i=1,2$ ).

The equations of motion for the masses when the pressures  $P_1, P_2$  act on the exposed faces are given by

$$\frac{d^2 x_i}{dt^2} + 2\omega_i \xi_i(x_i) \frac{dx_i}{dt} - \omega_i^2 \psi_i(x_i) = \{F_i - s_c(x_i, x_{3-i})\}/m_i, \\ \omega_i^2 = k_i/m_i, F_i = P_i l_g d_i, (i=1,2), \quad (1)$$

where  $\omega_i$  is the fundamental oscillating frequency,  $F_i$  is the force required to push the each cords up, and  $k_i$  is the linear stiffness. Although the coupling spring  $s_c$  has really an oblique force component, in the present work we neglected the depth between the masses, simply  $s_c(x_1, x_2) = k_c(x_1 - x_2)$ , where  $k_c$  is the linear stiffness.

## 3 The Exponential Spring and Damping

The purpose of the present synthesis model is to generate the chaotic behavior making the cords oscillation into nonlinear. We defined the nonlinear factors in eq. (1) by the proper equations.

We used the potential of nonlinear spring with constants  $A, a, b, c$  (the sign of  $a, b$  is assumed to be equal), that is,  $\phi(x) = A \exp(-bx) + ax + c$ . Especially in the case of  $A = ab$ , and  $\varepsilon_1 = (-a) \quad (-1/b) > 0$ , the restoring forces at the distances  $x_i$  ( $i=1,2$ ) are the derivatives with respect to  $x_i$  of these potentials therefore:

$$\psi(x_i) = -d\phi(x_i)/dx_i = \varepsilon_1 \{1 - \exp(x_i/\varepsilon_1)\}. \quad (2)$$

These forces required to produce the deflection are illustrated in Fig. 2. They are drawn near linear ( $\psi(x) \approx -x$ ) as  $\varepsilon_1$  increases, and the nonlinear property becomes stronger as  $\varepsilon_1$  decreases.

The nonlinear viscous losses are assumed piece-wise linear. They are caused to increase step-wise on the cords closure to represent the "soft stickiness". The collision is caused at the closure and has two kinds of cases, hard or viscous contact. For the hard boundary, the forces are immediately in a direction to open the port. For the viscous contact, however, the masses continue on to a displacement which exceeds  $x_c$ . We express these losses utilizing the constant damping rate  $\zeta$  and the exponential function:

$$\xi(x_i) = \zeta \exp\{-(x_i - x_c)/\varepsilon_2\}, \quad (3)$$

where  $\varepsilon_2$  is a positive constant. In Fig. 3, the point of intersection by the curves is the collision position ( $x_i = x_c$ ), and the damping rates become  $1+\zeta$  (critical damping). The losses for  $x_i < x_c$  can be largely changed by the control of  $\varepsilon_2$ .

#### 4 Equivalent Circuit and Forces Acting on the Cords

The acoustic impedance elements of the glottal orifice constitute the equivalent circuit as shown in Fig. 4. Since we assume the glottal flow to be in a quasi-steady state, these impedance elements can be expressed similar to Ishizaka's model<sup>7</sup> as  $\rho$  is the air density and  $\mu$  is the kinetic viscosity of air, i.e. eqs. (4),(5),(6). The vocal tract is represented in Fig. 4 as a transmission line of  $n$ -cylindrical, hard-walled sections. The element values applied to eq. (5) as the cross-sectional areas  $A_1 \cdots A_n$ , the cylinder length  $l_m$ , the sound velocity  $c$ , and the circumference of the  $j$ -th section  $S_j$ . The transmission line is terminated by a radiation load equal to that for a circular piston in an infinite baffle. In this model, the pressure acting on the radiation load  $R_o$  corresponds to the synthesized vocal sound.

$$\begin{aligned} R_c &= 1.37 R_{bb} |U_g| h_1^{-2}, R_e = -0.5 R_{bb} |U_g| h_2^{-2}, \\ R_{v_i} &= R_v d_i h_i^{-3}, L_{g_i} = L_g d_i h_i^{-1}, \\ R_{12} &= R_{bb} |U_g| (h_2^{-2} - h_1^{-2}), \quad (i = 1, 2), \end{aligned} \quad (4)$$

where  $R_{bb} = \rho/8l_g^2$ ,  $R_v = 3\mu/2l_g$ ,  $L_g = \rho/2l_g$ .

$$\begin{aligned} R_j &= S_j A_j^{-2} \sqrt{\rho \mu \omega} / 2 \cdot L_j = 0.5 l_j \rho A_j^{-1}, \\ C_j &= l_j d_j / \rho c^2, \quad (j = 1, \dots, n). \end{aligned} \quad (5)$$

$$R_o = 128 c \rho / 9 \pi^2 A_o \cdot L_o = 8 \rho / 3 \pi \sqrt{\pi A_o} \quad (6)$$

Then the pressures acting on the cords become the forces to oscillate the masses. The pressures at opening periods ( $h_1, h_2 > 0$ ) are defined to solve the circuit equations:

$$\begin{aligned} P_1 &= P_s - (R_c + R_{v1}/2) U_g - L_{g1}/2 \cdot dU_g/dt, \\ P_2 &= P_1 - \{R_{12} + (R_{v1} + R_{v2})/2\} U_g + (L_{g1} + L_{g2})/2 \cdot dU_g/dt, \end{aligned} \quad (7)$$

and subglottal pressure acts on the closure surface at the closure, i.e.  $P_1, P_2 = P_s$  at  $h_1 > 0$  and  $h_2 \leq 0$  or  $P_1 = P_s, P_2 = 0$  at  $h_1 \leq 0$ .

#### 5 Digital Simulation

We simulate the present synthesis model utilize the solution of the ordinary differential equations which solved by Runge-Kutta method.

First the differential equation of the volume velocity  $U_g$  is computed by the circuit equations. However because of the elements in the circuit can't be defined when the cords are closure ( $h_1$  or  $h_2 \leq 0$ ), we express the differential equation to stop the volume velocity theoretically:  $dU_g/dt = -U_g/\Delta T$ , where  $\Delta T$  is very little time interval for Runge-Kutta method.

Next the differential equations of the volume velocities  $U_j$ ,  $U_o$  and the pressures  $P_j$  ( $j = 1, \dots, n$ ) are similar computed by the circuit equations. And the differential equations of the displacements  $x_i$  ( $i = 1, 2$ ) are solved by the equations of motion for the masses. To continue these digital simulation as the time interval  $\Delta T$ , the vocal sound wave in each moment is produced.

The constants used in this simulation is shown in Table 1. In this simulation, interval  $\Delta T$  is  $10^{-6}$  [s], sampling frequency is 44[kHz], the cross-sectional areas in the tract are utilized the area function from the X-ray picture by Fant<sup>9</sup>.

#### 6 Result of Simulation and Consideration

The cross-sectional area of the cords and the synthesized vocal sound as vowels /a/, /i/, /u/ are showed in Fig. 5. The results show that the phase difference exists between area waves and it may be compared well with observations which have been made on the human vocal cords.  $A_{g2}$  shows the remarkable fluctuation for vowel /u/. It is considerable that this fluctuation affects nonlinearly on the output waves.

The waves makes up the characteristic wave in the third period with little variation, and the waves for /a/, /i/ have the fluctuations clearly. Therefore the

fluctuation is found to be affected by the tract shape, and is excepted that the chaotic properties vary similarly.

Next to examine the self-similarity we embedded the time series  $f(t)$  of the sound data into the three-dimensional space as  $(x, y, z) = (f(t), f(t-\tau), f(t-2\tau))$ , where  $\tau$  is the delay time determined from the first zero point of the auto-correlation function. In this way we composed the 3 kinds of attractors, (1) the human voices, (2) the synthesized voices by Ishizaka, et al.<sup>7</sup>, and (3) the synthesized voices by the present model. The human voices were sampled at 44[kHz] with 16[bits] by a sound board (Canopus Corp.: AUDIOPAQ-98). The other voices were sampled over the continuous 3,000 [steps] data after 1.5[sec] from beginning of the synthesis (as almost stable).

The attractors in the x-y plane for each vowels are showed in Fig. 6. In the human vowels of each pictures, the peculiar fluctuations caused by the chaotic phenomenon appear like the band. This band corresponds to the chaotic feature to express the intensity of the freedom. Also for the Ishizaka's model the bands are partly found, because this model includes the nonlinear factor. However band structures are distributed into the limited parts, and the phenomenon which the attractors leap on the periodic waves can be observed. As against these results, although the attractors in the present model resemble the track for another model, we can judge that the chaotic property for this model is more clearer than another model because of all attractors become the band. Since this band seems to be thin in comparison with the human voices, therefore this synthesized voices have less fluctuations.

Next we show the Lyapunov analysis used the continuous 44,000 [steps] data, for the human and synthesized voices, and examined the degrees of chaos. Since the measurements of the maximum Lyapunov exponents shown in Table 2 become all positive, consequently the orbital instability exists. But these values were lower than those of the human vocal sounds, by 1/5 ~ 1/3. Furthermore the Lyapunov dimensions shown in Table 3 become all non-integer and close to the values of the human voices for /a/. Since for /i/, /u/, however, they were low about 0.4, the self-similarity were insufficient for /i/, /u/. Therefore these chaotic properties were considered to be insufficient.

At last we consider the hearing examination with audio equipment. These results were that the synthesized vocal sounds could be heard with more

humanity than the voices by another model, which seemed to be unnatural. We can consider this factor is caused not only by the insufficiency of the chaotic properties but also by the subglottal pressure  $P_s$  treated as a constant.

## 7 Conclusion

In the present work, we involved the nonlinear factors into the traditional model, and synthesized the vocal sounds with the chaotic properties by the nonlinear cord oscillations. Also the synthesized voices were found to be the chaotic properties through the Lyapunov analysis, where as the chaotic properties were insufficiency.

As the subject in the future, the effort to increase the orbital instability and to get closer to the human vocal sounds will be required.

## References

1. Lorenz EN(1963), Deterministic nonperiodic flow, J.Atoms.Sci.20, pp 130-141.
2. Mandelbrot BB(1982), Fractal Geometry of Nature, Freeman, San Francisco.
3. Yamaguchi T, Nagano M, Nakagawa M(1993-1), A Study of Fractal Properties of Vocal Sounds (Fractal Dimensions of Vowels), Technical Report of IEICE, SP92-130, pp 757-64.
4. Sabanal S, Nakagawa M(1996), The Fractal Properties of Vocal Sounds and Their Application in the Speech Recognition Model, Chaos, Solitons & Fractals Vol.7, No. 11, pp 1825-1843.
5. Flanagan JL, Landgraf L(1968), Self-oscillating source for vocal-tract synthesizers, IEEE Trans. Audio and Electro-acoust., AU-16, pp 57-64.
6. Flanagan JL(1972), Speech Analysis Synthesis and Perception, 2nd ed. pp 246-259, Springer, New York.
7. Ishizaka K, Flanagan JL(1972), Synthesis of Voiced sounds From a Two-Mass Model of the Vocal Cords, Bell System Tech.J., 51, 6, pp 1233-1268.
8. Yamaguchi T, Masahiro N(1993-10), Fractal Property of Vocal Sounds and Its Evaluation Model, Technical Report of IEICE, SP93-74, DSP93 -75, pp 79-86.
9. Fant(1960), Acoustic theory of speech production's-Gravenhage:Mouton & Co.

Table 1 : Variables used in the present model

	Meaning	Value	Unit
$m_1$	mass 1 of vocal cords	0.125	g
$m_2$	mass 2 of vocal cords	0.025	g
$d_1$	thickness of mass 1	0.25	cm
$d_2$	thickness of mass 2	0.05	cm
$l_g$	length of glottis	1.4	cm
$k_1$	linear spring coeff.	80000	dyn/cm
$k_2$	linear spring coeff	8000	dyn/cm
$k_c$	coupling spring coeff.	25000	dyn/cm
$\varepsilon_1$	nonlinear spring coeff.	0.01	—
$\varepsilon_2$	nonlinear damp. coeff.	0.00002	—
$\rho$	density of air	$1.14 \times 10^{-3}$	g/cm <sup>3</sup>
$\mu$	viscosity of air	$1.86 \times 10^{-4}$	dyn s /cm <sup>2</sup>
$c$	sound velocity	$3.5 \times 10^4$	cm/s
$\zeta_1$	damping resistance	0.1	—
$\zeta_2$	damping resistance	0.6	—
$P_s$	subglottal pressure	8	cmH <sub>2</sub> O*

\* 1 [cmH<sub>2</sub>O] = 980.7 [dyn/cm<sup>2</sup>]

Table 2: The maximum

Lyapunov exponents [bit]

Vowels	Human Voices	Synthesized Voices
/a/	0.099	0.032
/i/	0.065	0.013
/u/	0.029	0.006

Table 3: The Lyapunov dimensions

Vowels	Human Voices	Synthesized Voices
/a/	2.21	2.24
/i/	2.16	1.79
/u/	1.88	1.50

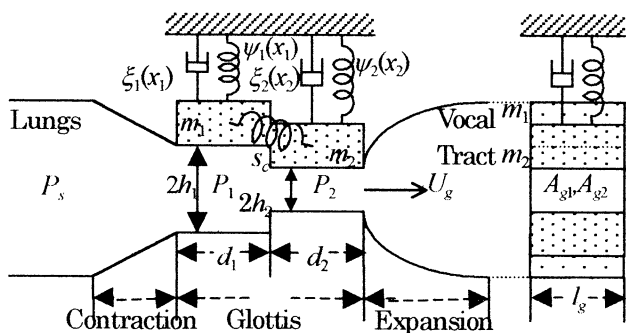


Fig. 1 Schematic diagram of the two-mass approximation of the vocal cords.

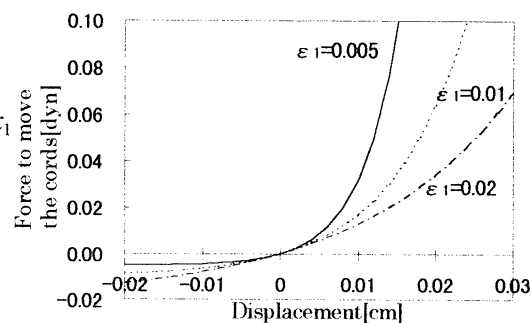


Fig. 2 Characteristics of the exponential stiffnesses.

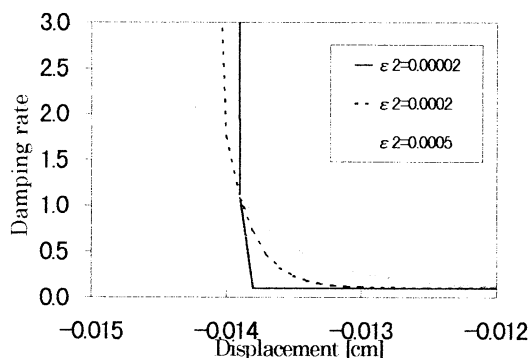


Fig. 3 Characteristics of the exponential dampings (in the case of  $\zeta=0.1$ ).

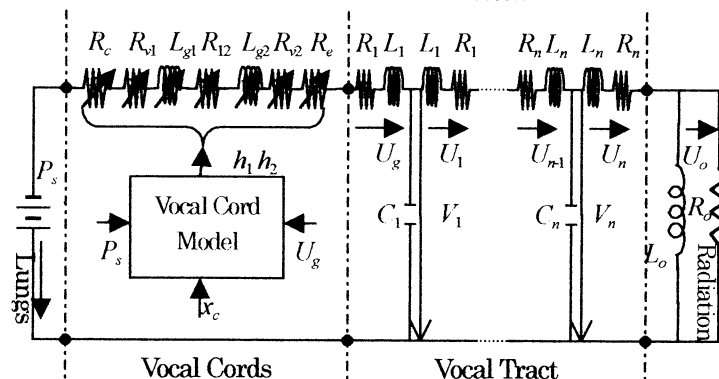


Fig. 4. Network model for the synthesis of voiced sounds.

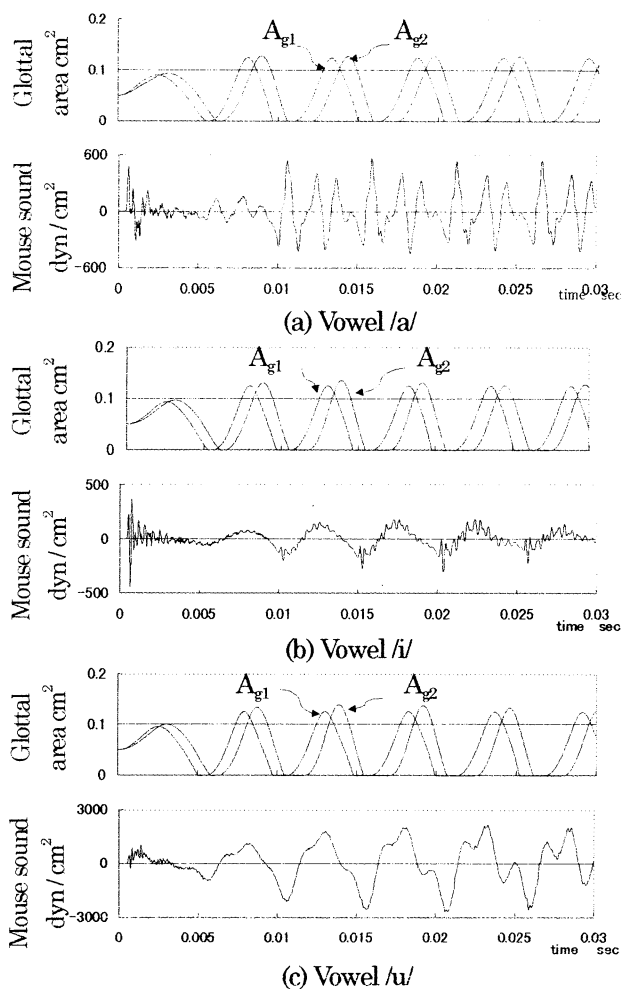


Fig.5 The results of the simulation for vowels /a/, /i/, /u/.

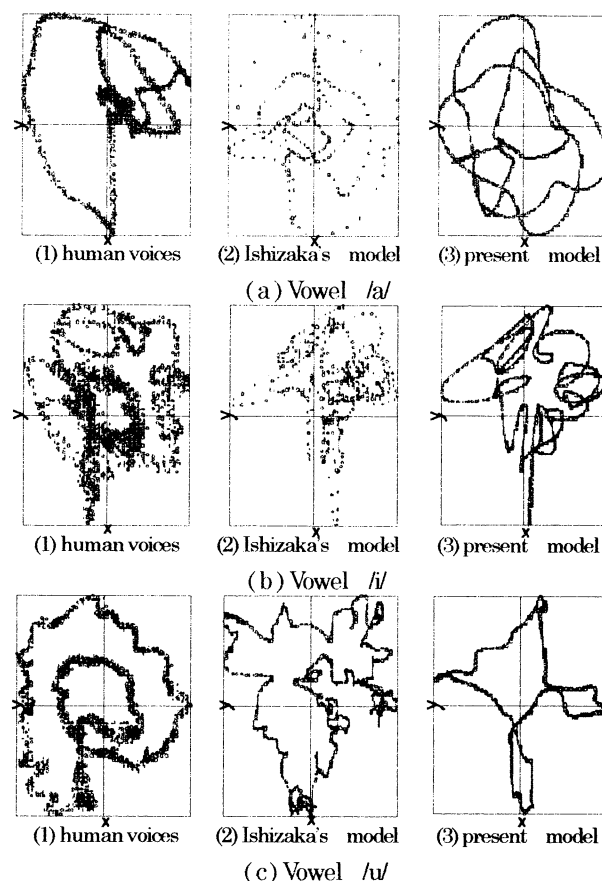


Fig. 6 The attractors of vocal sounds for vowels /a/, /i/, /u/.

This is including (1) human vocal sounds, (2) synthesized voices by Ishizaka's model, and (3) synthesized voices by the present model.



# A Chaos Model to Solve the Optimal Stable Marriage Problem

Ryuichi Hiroi

Masahiro Nakagawa<sup>†</sup>

Department of Electrical Engineering, Faculty of Engineering, Nagaoka University of Technology  
Kamitomioka 1603-1, Nagaoka, Niigata, 940-2188. Tel.0258-47-1611(Ext.5476)

hiroii@pelican.nagaokaut.ac.jp

<sup>†</sup>masanaka@vos.nagaokaut.ac.jp

## Abstract

In this paper, we propose a chaos neural network model to solve the optimal stable marriage problem. The purpose of this problem is to search matching of men and women such that the total satisfaction of all people takes a maximum value in the stable matching. First we examined a non-chaos neural network model to solve this problem. Second we considered a chaos neural network model to overcome a shortcoming that the network falls into a local minimum of energy. In addition the performances of both models were compared. In conclusion, chaos neural network model is found to have a good performance, then we also confirmed that an introduction of chaos dynamics is significant.

## Keyword

Neural Network, Chaos Dynamics, Stable Marriage Problem, Optimal Stable Matching.

## 1 Introduction

The stable marriage problem was introduced by Gale and Shapley<sup>1</sup> in 1962. The purpose of this problem is to search the stable matching of men and women. Then they proposed solution of men (women) priority. But there is a disadvantage that the results become best for men (women) and worst for women (men) in these solutions. To overcome this disadvantage, various extended stable marriage problems<sup>2,3,4</sup> have been studied to date. The optimal stable marriage problem is considered to be a sort of them. We propose in this paper a chaos neural network model to solve the optimal stable marriage problem.

## 2 The definition of the optimal stable marriage problem

For the optimal stable marriage problem of size  $N$ , each  $N$  men and  $N$  women rank the members of another sex in the order of preference. In such an order of preference, we search the optimal and the stable matching. The stable matching means matching of men and women such that there exists no man and woman who are not paired in which both of them prefer each

other as their actual partners under the matching. Especially the optimal stable matching means matching of men and women so as to maximize the total satisfaction of all people in the stable matching.

The preference list which represents the order of preference of men and women is defined by

$$m_{ij} = k \quad \text{if man } i \text{ is the } j\text{th choice of woman } k,$$

$$w_{ij} = k \quad \text{if woman } i \text{ is the } j\text{th choice of man } k.$$

Similarly the satisfaction list is defined by

$$a_{ij} = k \quad \text{if man } i \text{ is the } k\text{th choice of woman } j,$$

$$b_{ij} = k \quad \text{if woman } i \text{ is the } k\text{th choice of man } j.$$

We can derive uniquely the satisfaction list from the preference list.

Suppose that, for a given stable marriage,

$$S = \{(M_1, W_1), (M_2, W_2), \dots, (M_N, W_N)\}, \quad (1)$$

is a stable matching herewith  $M_i$  and  $W_i$  are paired. We define  $c(S)$  which means the total satisfaction of all people.

$$c(S) = cm(S) + cw(S),$$

$$cm(S) = \sum_{i=1}^N a_{(M_i)(W_i)} \quad \text{men only}, \quad (2)$$

$$cw(S) = \sum_{i=1}^N b_{(W_i)(M_i)} \quad \text{women only}.$$

Therefore the optimal stable matching is the stable matching such that  $c(S)$  becomes a minimum.

We show some examples of the preference list, the satisfaction list and the stable matching in Tables 1, 2 and 3. The optimal stable matching is  $S_2$  in this example.

## 3 The neural network approach

The neurons are allotted by the checkerboard. If the output  $x_{ij}$  of neuron of  $i$  row and  $j$  column takes 1, then man  $i$  and woman  $j$  are paired. On the other hand,  $x_{ij}$  has 0, then man  $i$  and woman  $j$  is not paired. We show an example of the allocation of neurons in Figure 1.

Table 1 The preference list

$m_{ij}$	$w_{ij}$
1: 5 2 8 6 4 1 7 3	1: 2 4 5 8 3 7 1 6
2: 1 4 3 6 2 7 8 5	2: 7 4 1 6 3 5 2 8
3: 2 5 7 4 3 1 8 6	3: 1 3 7 4 8 2 5 6
4: 3 4 5 1 8 2 7 6	4: 8 6 4 7 1 3 5 2
5: 1 5 6 7 4 3 8 2	5: 4 3 1 6 5 2 8 7
6: 4 1 8 5 7 6 3 2	6: 3 1 7 5 6 2 4 8
7: 1 7 5 4 6 2 8 3	7: 4 6 5 7 8 1 3 2
8: 4 6 8 5 7 3 1 2	8: 5 2 6 1 8 4 7 3

Table 2 The satisfaction list

$a_{ij}$	$b_{ij}$
1: 6 2 8 5 1 4 7 3	1: 7 1 5 2 3 8 6 4
2: 1 5 3 2 8 4 6 7	2: 3 7 5 2 6 4 1 8
3: 6 1 5 4 2 8 3 7	3: 1 6 2 4 7 8 3 5
4: 4 6 1 2 3 8 7 5	4: 5 8 6 3 7 2 4 1
5: 1 8 6 5 2 3 4 7	5: 3 6 2 1 5 4 8 7
6: 2 8 7 1 4 6 5 3	6: 2 6 1 7 4 5 3 8
7: 1 6 8 4 3 5 2 7	7: 6 8 7 1 3 2 4 5
8: 7 8 6 1 4 2 5 3	8: 4 2 8 6 1 3 7 5

Table 3 The stable matching

	$cm(S)$	$cw(S)$	$ cm(S) - cw(S) $	$c(S)$
$S_0 = \{(1,5), (2,1), (3,2), (4,3), (5,6), (6,8), (7,7), (8,4)\}$	13	25	12	38
$S_1 = \{(1,5), (2,1), (3,2), (4,3), (5,7), (6,8), (7,6), (8,4)\}$	17	23	6	40
$S_2 = \{(1,2), (2,1), (3,5), (4,3), (5,6), (6,8), (7,7), (8,4)\}$	15	22	7	37
$S_3 = \{(1,5), (2,1), (3,2), (4,3), (5,8), (6,7), (7,6), (8,4)\}$	22	20	2	42
$S_4 = \{(1,2), (2,1), (3,5), (4,3), (5,7), (6,8), (7,6), (8,4)\}$	19	20	1	39
$S_5 = \{(1,2), (2,1), (3,3), (4,5), (5,6), (6,8), (7,7), (8,4)\}$	20	19	1	39
$S_6 = \{(1,2), (2,1), (3,5), (4,3), (5,8), (6,7), (7,6), (8,4)\}$	24	17	7	41
$S_7 = \{(1,6), (2,1), (3,5), (4,3), (5,7), (6,8), (7,2), (8,4)\}$	22	17	5	39
$S_8 = \{(1,2), (2,1), (3,3), (4,5), (5,7), (6,8), (7,6), (8,4)\}$	24	17	7	41
$S_9 = \{(1,6), (2,1), (3,5), (4,3), (5,8), (6,7), (7,2), (8,4)\}$	27	14	13	41
$S_{10} = \{(1,6), (2,1), (3,3), (4,5), (5,7), (6,8), (7,2), (8,4)\}$	27	14	13	41
$S_{11} = \{(1,2), (2,1), (3,3), (4,5), (5,8), (6,7), (7,6), (8,4)\}$	29	14	15	43
$S_{12} = \{(1,3), (2,1), (3,6), (4,5), (5,7), (6,8), (7,2), (8,4)\}$	34	12	22	46
$S_{13} = \{(1,6), (2,1), (3,3), (4,5), (5,8), (6,7), (7,2), (8,4)\}$	32	11	21	43
$S_{14} = \{(1,5), (2,1), (3,2), (4,3), (5,6), (6,8), (7,7), (8,4)\}$	39	9	30	48

The present neural network is defined by the energy function as follows.

$$\begin{aligned}
E = & \frac{A_1}{2} \sum_{i=1}^N \left( \sum_{l=1}^N x_{il} - 1 \right)^2 + \frac{A_2}{2} \sum_{j=1}^N \left( \sum_{k=1}^N x_{kj} - 1 \right)^2 \\
& + \frac{B}{2} \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^N \sum_{l=1}^N d_{ijkl} x_{ij} x_{kl} + \frac{O}{2} \sum_{i=1}^N \sum_{j=1}^N s_{ij} x_{ij}^2 \quad (3) \\
& + \frac{D}{2} \sum_{i=1}^N \sum_{j=1}^N x_{ij} (1 - x_{ij}).
\end{aligned}$$

Here  $A_1$ ,  $A_2$ ,  $B$ ,  $O$  and  $D$  are adjustable parameters of the energy function. The first and the second terms correspond to the constraints such that all people must be paired with only one. The third term is the cost function for the stable matching. The fourth term is the cost function for the satisfaction to be maximized. The final term is the cost function such that  $x_{ij}$  prefers 1 or 0.

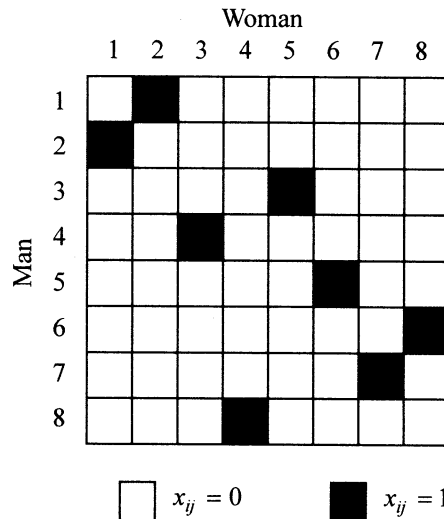


Figure 1 The allocation of neurons

In addition  $s_{ij}$  and  $d_{ijkl}$  are defined by

$$s_{ij} = a_{ij} + b_{ji} , \quad (4)$$

$$d_{ijkl} = f(a_{il}, a_{ij}) \cdot f(b_{li}, b_{lk}) + f(a_{kj}, a_{kl}) \cdot f(b_{jk}, b_{ji}) ,$$

$$f(\alpha, \beta) = \begin{cases} 1 & \text{if } \alpha < \beta \\ 0 & \text{otherwise} \end{cases} . \quad (5)$$

Here we define  $d'_{ijkl}$  as follows.

$$d'_{ijkl} = \begin{cases} d_{ijkl} & \text{if } i \neq k \text{ and } j \neq l \\ \frac{O}{B} s_{ij} & \text{if } i = k \text{ and } j = l \\ 0 & \text{otherwise} \end{cases} . \quad (6)$$

Accordingly the energy function  $E$  can be rewritten from the function (3) according to the function (6).

$$E = \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \sum_{k=1}^N \sum_{l=1}^N W_{ijkl} x_{ij} x_{kl} + \sum_{i=1}^N \sum_{j=1}^N \Theta_{ij} x_{ij} + C ,$$

$$W_{ijkl} = A_1 \delta_{ik} + A_2 \delta_{jl} + B d'_{ijkl} - D \delta_{ik} \delta_{jl} , \quad (7)$$

$$\Theta_{ij} = \frac{-2A_1 - 2A_2 + D}{2} ,$$

$$C = \frac{A_1 + A_2}{2} N .$$

In addition the expressions to derive the output  $x_{ij}$  and to update the internal state  $y_{ij}$  of the neuron are defined by

$$x_{ij}(t+1) = \frac{1}{1 + \exp\left(-\frac{y_{ij}(t+1)}{\varepsilon}\right)} , \quad (8)$$

$$y_{ij}(t+1) = y_{ij}(t) - h \frac{\partial E(t)}{\partial x_{ij}(t)}$$

$$= y_{ij}(t) + h \left\{ - \sum_{k=1}^N \sum_{l=1}^N W_{ijkl} x_{kl}(t) - \Theta_{ij} \right\} . \quad (9)$$

To date, however, the neural network solution has encountered a shortcoming such that the network state falls into a local minimum of the energy. We call the above model the non-chaos model in this paper.

#### 4 The chaos neural network approach

Let us consider a chaos neural network model<sup>5,6</sup> to overcome the previously mentioned shortcoming.

We define the expression to update the internal state  $y_{ij}$  of the chaos neuron by the following.

$$y_{ij}(t+1) = k y_{ij}(t) + h \left\{ - \sum_{k=1}^N \sum_{l=1}^N W_{ijkl} x_{kl}(t) - \Theta_{ij} \right\} - \alpha x_{ij}(t) + s , \quad (10)$$

where  $k$  is a memory constant,  $\alpha$  is a refractory

constant and  $s$  is an external stimulus.

In addition we define the control strategy of each parameters by the followings.

$$\tau(t+1) = \begin{cases} \tau(t) - \kappa \tau(t) \{1 - \tau(t)\} & : \tau(t) \geq \tau_c \\ \tau_n & : \tau(t) < \tau_c \end{cases} ,$$

$$\varepsilon(t) = \varepsilon_n \tau(t) + \varepsilon_c \{1 - \tau(t)\} ,$$

$$k(t) = k_n \tau(t) + k_c \{1 - \tau(t)\} , \quad (11)$$

$$\alpha(t) = \alpha_n \tau(t) + \alpha_c \{1 - \tau(t)\} ,$$

$$s(t) = s_n \tau(t) + s_c \{1 - \tau(t)\} ,$$

$$h(t) = h_n \tau(t) + h_c \{1 - \tau(t)\} .$$

$\varepsilon_n$ ,  $k_n$ ,  $\alpha_n$ ,  $s_n$  and  $h_n$  are the values of each parameters in non chaos model.  $\varepsilon_c$ ,  $k_c$ ,  $\alpha_c$ ,  $s_c$  and  $h_c$  are the values of each parameters in chaos model. The time-dependence of the variable  $\tau$  is shown in Figure 2.

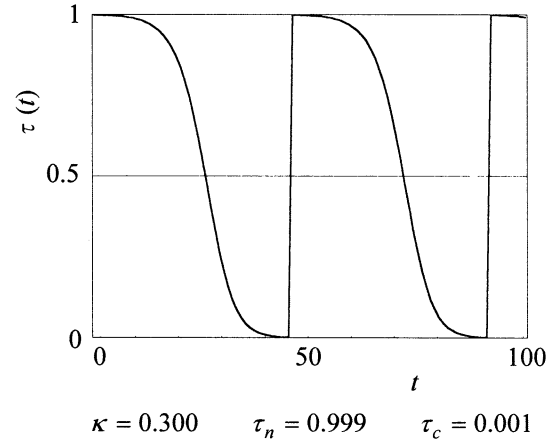


Figure 2  $\tau$  vs.  $t$

We call the above model the chaos model in this paper.

#### 5 Results of simulation and discussions

Let us compare the performance of the chaos model with that of the non-chaos model with size 8.

First we prepared 100 random preference lists, and searched for the optimal stable matching in them by the rotation digraph<sup>2</sup>. Next we tried 10 times simulation in one preference list. Consequently we evaluated the finding rate of the optimal stable matching and the mean value of the updating epochs. Initialization of  $y_{ij}$  and each parameters are given in Table 4.

We show the finding rate of the optimal stable matching and the mean value of updating epochs in table 5. The rate of errors different from the searched matching and the optimal stable matching are given in Figure 3.

The chaos model was found to increase the mean value of the updating epochs, whereas it becomes better

for the finding rate than non-chaos model. Here, evaluating the averaged updating epochs for 86 preference lists within 300 epochs, we find the value of 50.3 at most. In addition even though the present model was not successful to find the optimal stable matching, the rate of error is found to be less than 5%. Therefore it may be concluded that an introduction of chaos dynamics in this optimization problem is effective.

Table 4 Conditions of simulation

	Non-chaos	Chaos
Initialization of $y_{ij}$	Within $\pm 50\%$ of $\varepsilon$	Within $\pm 50\%$ of $\varepsilon_n$
$A_1$	1.000	1.000
$A_2$	1.000	1.000
$B$	1.000	1.000
$O$	0.100	0.100
$D$	1.000	1.000
$\varepsilon$	0.300	————
$h$	1.000	————
$\varepsilon_n$	————	0.300
$\varepsilon_c$	————	0.020
$k_n$	————	1.000
$k_c$	————	0.800
$\alpha_n$	————	0.000
$\alpha_c$	————	1.000
$s_n$	————	0.000
$s_c$	————	0.400
$h_n$	————	1.000
$h_c$	————	0.050
$\kappa$	————	0.300
$\tau_n$	————	0.999
$\tau_c$	————	0.001

Table 5 The finding rate and the mean value of updating epochs

	Non-chaos	Chaos
The finding rate	67%	98%
The mean value of the updating epochs	16.2	184.4
The averaged updating epochs for preference lists within 300 epochs	————	50.3

## 6 Conclusions

In this report, we have considered the chaos neural network model to solve the optimal stable marriage problem. Consequently the mean value of updating epochs becomes larger, whereas the rate to find the optimal stable matching becomes large in the chaos model. However several parameters are to be carefully chosen in the chaos model.

In the future, it seems to be worthwhile to consider the parameter controlled chaos neural network in this problem as previously investigated. Then it may be also fascinating subject to construct a parallel processing algorithm to solve this kind of optimization problem.

## References

1. Gale D, Shapley LS (1962), College admissions and the stability of marriage. Am. Math. Monthly, Vol.69, pp.9-15.
2. Irving RW, Leather P, Gusfield D (1987), An Efficient Algorithm for the "Optimal" Stable Marriage. J.ACM, Vol.34, No.3, pp.532-543
3. Nakamura M, Onaga K, Kyan S (1994), Sex-Fair Stable Marriage Problem and Its GA Solution. IEICE Trans. Fundamentals, Vol.E78-A, No.6, pp.664-670
4. Yuan Y, Wang L (1996), A neural network approach to solve the stable matching problem. European.J Operation Research, Vol.93, No.2, pp.331-345
5. Yamada T, Aihara K, Kotani M (1993), Chaotic Neural Networks and The Traveling Salesman Problem. Proceedings of 1993 International Joint Conference on Neural Networks, pp.1549-1552.
6. Nakagawa M (1996), A Parameter Controlled Chaos Neural Network. Journal of the Physical Society of Japan, Vol.65, No.6, pp.1859-1867.

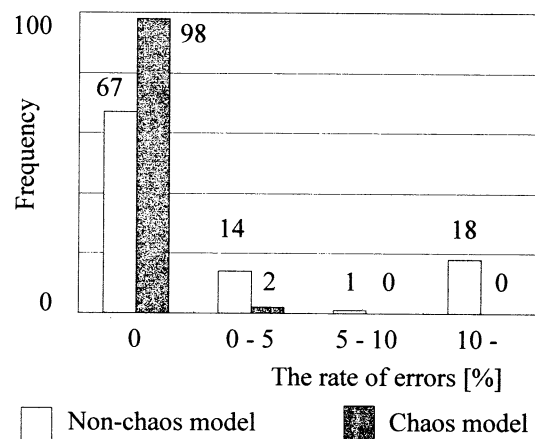


Figure 3 The rate of errors

## Computation of Motion Direction by Output Neurons of the Retina

Hiroyuki Uchiyama

Department of Information and Computer Science,  
Faculty of Engineering, Kagoshima University,  
Kagoshima 890-0065, Japan  
uchiyama@ics.kagoshima-u.ac.jp

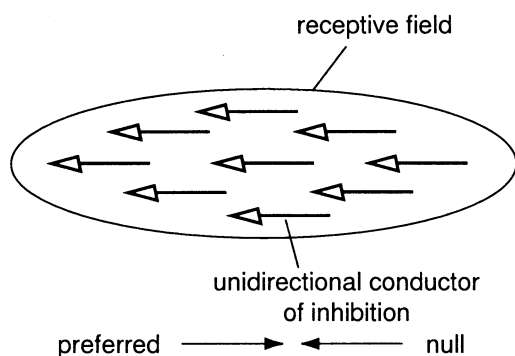
### Abstract

One type of retinal output neurons (= ganglion cells) prefers object motion in a particular direction. Neuronal mechanisms for the computation of motion direction are still unknown. We quantitatively mapped excitatory and inhibitory regions of receptive fields for directionally selective retinal ganglion cells in the Japanese quail, and found that the inhibitory regions are displaced about 1-3 deg. toward the side where the null sweep starts, relative to the excitatory regions. Directional selectivity thus results from delayed transient suppression exerted by the nonconcentrically-arranged inhibitory regions, and not by unidirectional inhibitory conduction as hypothesized by Barlow and Levick.

**Key words:** visual system, motion perception, directional selectivity, retinal ganglion cells, electrophysiology

### Introduction

Animals must process dynamic visual information, because they live in dynamic interaction with their environment<sup>1</sup>. Directionally selective (DS) visual neurons prefer object motion in a particular direction and are thought to have an essential role for computation of visual motion information<sup>2-4</sup>. While many animals are known to possess a rich population of DS retinal output neurons (= retinal ganglion cells, RGCs)<sup>5-11</sup>, it is still unclear how their directional selectivity is computed from motion-less information processed by distal retinal neurons.



**Fig. 1** Barlow-Levick model for retinal directional selectivity.

Barlow and Levick<sup>12</sup> proposed that unidirectional inhibitory conduction gives rise to directional selectivity in the retina (Fig. 1), and computational models have been constructed based on this hypothesis<sup>13,14</sup>. While starburst amacrine cells have been proposed as a possible neural substrate for unidirectional inhibitory conduction<sup>15,16</sup>, very recently it has been shown that directional selectivity remains unchanged in the rabbit DS RGCs after selective destruction of starburst amacrine

cells<sup>17</sup>. Starburst amacrine cells may potentiate the responses of RGCs to moving stimuli, but the potentiation does not appear to have directional bias<sup>17,18</sup>. Thus it remains unknown how unidirectional inhibitory conduction is implemented for DS RGCs. Is unidirectional inhibitory conduction essential for directional selectivity? We report here nonconcentric receptive field (RF) organization of DS RGCs in the Japanese quail, and conclude that this organization is a basis for directional selectivity.

### Methods

Sixty-eight Japanese quail (*Coturnix japonica*) of both sexes were used in the present study. Animals were treated in accordance with the animal usage guideline of the Society for Neuroscience. A tungsten electrode (12 M $\Omega$ ; A-M Systems, Everett, WA) was inserted stereotactically into the optic chiasm through the hole of the skull. Nerve impulses from single optic nerve fibers were isolated, amplified, detected with a time-window discriminator, and fed to an intelligent computer interface (CED 1401plus, Cambridge Electronic Design, England) and an IBM-AT-compatible personal computer for analysis. Upon a successful isolation of single optic nerve fibers, we first made a rough mapping of the RF on a tangent screen using a flash light or a laser pointer. Then, we positioned the face of a monitor scope (Tektronix 608; CRT surface 10 x 12.5 cm) in the region of the RF at a distance of 20-60 cm from the cornea. Presentation of stimuli were controlled by Picasso Image Synthesizer (Innisfree, England), CED-1708 (Cambridge Electronic Design, England) and an IBM-AT-compatible personal computer.

RFs of the RGCs were quantitatively mapped using brief flashes (100-300 ms) of a small spot of light (0.5-1.2 deg. square; 10 cd/m<sup>2</sup>). The spots were presented at one of 9 by 11 lattice points on the monitor in a random

order. Evoked spikes were counted for 200-500 ms. Counts of all the sampling duration were used for two-dimensional RF mapping. Five to 10 responses to each point were averaged. Data sets of responses to the 99 points were interpolated with a cubic convolution method<sup>19</sup> or curvefitted with a two-dimensional Gaussian function.

Directional preference was measured with a light spot (0.5-1 deg. square, 16-25 deg./s) moving across the center of the RF in 12 directions. Responses for every 12 directions were curvefitted with a Gaussian function, and preferred directions were determined by the fitted function.

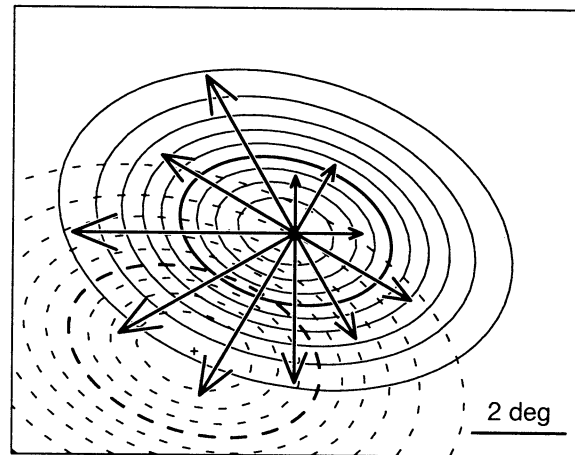
To measure spatial distribution of suppression, paired spots of light were presented; one (excitation spot) for the RF center, and another (suppression spot) for the periphery. The suppression stimulus led the excitation stimulus by 100 ms. Both the excitation and suppression spots were presented for 200-300 ms. While the excitation spot was always presented at center of the RFs, the leading suppression spot was presented at one of 9 by 11 lattice points on the monitor in a random order. Each presentation was repeated 10 times. Decreased amounts of on-off transient responses (30-50 ms duration) evoked by the excitation spot at 9 by 11 points were measured, and data sets of the 99 points were curvefitted with a 2-dimensional Gaussian function.

## Results

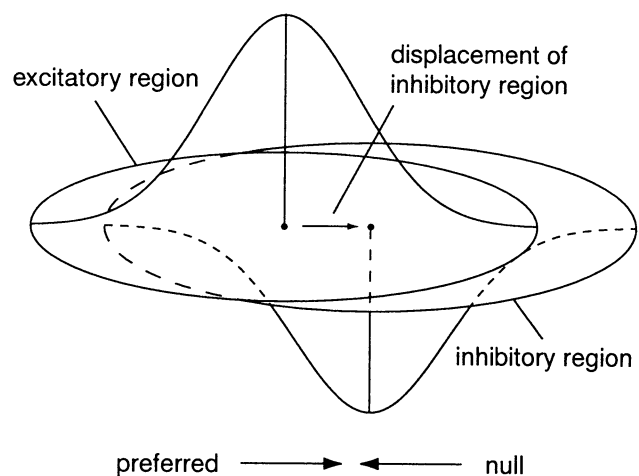
We mapped two-dimensional spatial structures of excitatory regions (ERs) and inhibitory regions (IRs) of the RFs of DS RGCs. Two-dimensional spatial sensitivity distributions of ERs were quantitatively measured in 118 RGCs<sup>20-22</sup>, and then their directional preference was examined. Among them, 33 units (28%) were identified as DS RGCs. Twenty-one units showed transient on and off responses to stationary spots of light and 12 units showed only transient off responses, among the 33 DS units. On-off DS RGCs showed responses with twin peaks to light spots moving in intermediate velocities, indicating that both the leading off-on and the trailing on-off transitions are effective. Off DS RGCs showed single-peak responses to moving stimuli. Although DS RGCs were clearly distinguishable from non-DS RGCs by responses to moving stimuli, responsivity to stimuli moving in the direction opposite to the preferred direction ("null" direction) varied. Some DS RGCs did emit almost no responses to the stimuli moving in the null direction, but the others showed some modest response to those stimuli. Thus non-DS motion sensitivities varied from unit to unit. Although the directional preferences of cells can not be predicted from the 2-dimensional structures of ERs, some neurons with elongated ERs have a preferred-null axis that is perpendicular to the longer axis of the ER.

Responses evoked by a stationary stimulus applied to the RFs were strongly suppressed by a stationary stimulus applied to the null side (After He and Masland<sup>17</sup>, the null side is defined as the side where stimulus movement in the null direction starts, and vice versa), but not strongly suppressed by a stimulus applied to the preferred side of the RF. The suppression was transient,

and the suppression spot, when turned on 50-100 ms before the excitation spot, maximally suppressed the response to the excitation spot. DS suppression was measured as differences between suppressions by stimuli applied to two sides of the RF along the preferred-null axis, and was transient and the largest at 100-ms time lag.



**Fig. 2** IR of a quail DS RGC (#5116) measured by paired stationary spots for excitation and suppression. ER (solid line) and IR (dotted line). Thick contour lines indicate the contours of 1-standard-deviation from the Gaussian-fitted centers. Other contour lines indicate 10%-step lines of the maximal response. Center of the IR is indicated by a white +. Compass plot indicates the directional tunings to a light spot moving in 12 directions. Length of the arrows is proportional to the responses.



**Fig. 3** Nonconcentric receptive field model for retinal directional selectivity.

For 11 DS RGCs (9 on-off units, 2 off unit), two-dimensional spatial distributions of the DS suppression were measured with paired stationary stimuli for excitation and suppression (Fig. 2). Although the suppression spot near the ER activated responses by themselves, the responses were transient and easily distinguishable from responses to the excitation spot. Fig. 2 shows one example of the Gaussian-fitted ER and IR. Although both the ERs and IRs were well fitted with a two-dimensional Gaussian function, the IRs were slightly asymmetrical along the preferred-null axis in some units: measured peaks shifted to a lesser extent

toward the ERs, relative to fitted peaks. Contours of 1-standard-deviation from the fitted centers of ERs and IRs were  $3.4 \pm 1.0$  (mean  $\pm$  S.D.,  $n = 11$ ) and  $3.4 \pm 0.8$  (mean  $\pm$  S.D.,  $n = 11$ ) deg. in diameter, respectively. Centers of the ERs and IRs were not aligned, and the distances between the centers were 1.2-3.2 deg. ( $2.0 \pm 0.6$  deg.,  $n = 11$ ). IRs were always shifted toward the null side of the cells (Fig. 2). Thus quail DS RGCs have nonconcentrically arranged RFs (Fig. 3). Estimation of displacements of the IRs based on the Gaussian-fitted ERs and IRs were significantly smaller in non-DS RGCs ( $0.4$ - $0.8$  deg.;  $0.6 \pm 0.2$  deg, mean  $\pm$  S.D.,  $n = 3$ ) than in DS RGCs.

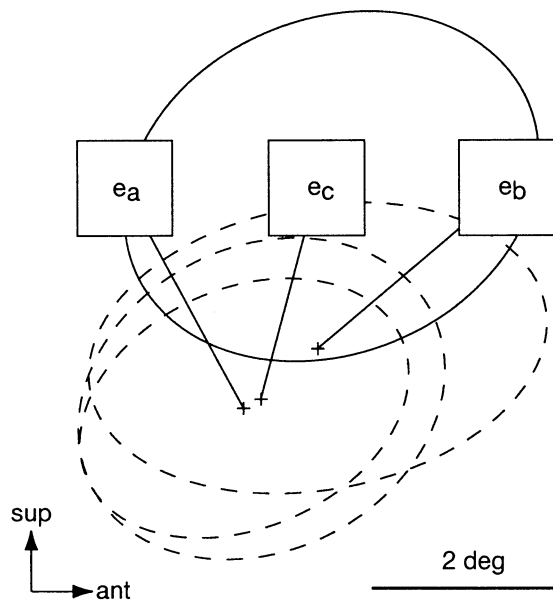


Fig. 4 Effect of excitation spot location on mappings of the IR of a quail DS RGC (#5176). The solid contour line indicates the ER (1-standard-deviation area). The dotted contour lines indicate the IRs (1-standard-deviation area), measured with three different excitation spots (solid squares,  $e_c$ ,  $e_a$ ,  $e_b$ ). Light spots were 1 deg. square ( $10 \text{ cd/m}^2$ ).

For a few DS RGCs, measurements of the IRs were repeated using three different locations of the excitation spot (Fig. 4). The hypothesis by Barlow and Levick<sup>12</sup> assumes that there are many local unidirectional conductors within the RFs, and that their directions of conduction are all the same (Fig. 1). Based on these assumptions, it may be expected that measured IRs are to be shifted according to excitation sites, if the excitation spot is relocated. But, that is not the case, as shown in Fig. 4. Size and location of the IRs were not much affected by location of the excitation spots (Fig. 4), and thus presence of local and unidirectional inhibitory conductors was not supported.

### Discussion

For a RGC with a nonconcentric RF, a moving object that stimulates the IR before the ER evoked a smaller response than an object moving on any other trajectory through the RF (Fig. 3). This trajectory defines the null direction. Stimulation in the opposite direction evokes a maximal response and defines the preferred

direction. Thus, displaced IRs may generate directional selectivity. Barlow and Levick<sup>12</sup> hypothesized that directional selectivity results from asymmetrical conduction of inhibition across the RF, based on their experiments on rabbit DS RGCs (Fig. 1). Wyatt and Daw<sup>23</sup> extended the Barlow-Levick model into two-dimensional space. However, when the IRs are displaced, as shown in the present study, such unidirectional conduction mechanisms may not be needed for directional selectivity. Furthermore, independence of the IR from excitation sites, as shown in Fig. 4, does not support the hypotheses of asymmetrical conduction of inhibition. Amthor and Grzywacz tried to map receptive field organization of rabbit DS RGCs quantitatively by means of paired stationary stimuli<sup>24,25</sup>. However, they chose practically one-dimensional stimuli, and failed to map the RFs two-dimensionally.

Although structural bases for the displacement of the IRs are unknown, the displacement of dendritic fields of rabbit DS RGCs reported by Yang and Masland<sup>26</sup> seems suggestive. They reported that dendritic fields were displaced toward the null side, relative to their RFs, in some rabbit DS RGCs. If distribution and density of inhibitory inputs on dendrites of DS RGCs are spatially biased toward the null side, relative to the distribution of excitatory inputs, displaced IRs could be achieved. Thus directional selectivity may be implemented as nonconcentrically-arranged distributions of excitatory and inhibitory inputs on the dendritic fields of DS RGCs. Biophysical mechanisms for delayed suppression maximally at around 100-ms-delay are also unknown. A GABA<sub>B</sub> agonist, baclofen, enhances directional selectivity of DS RGCs in a salamander<sup>27</sup>, and G-protein-coupled GABA<sub>B</sub> receptors mediate slow increase in potassium conductance, or late IPSPs<sup>28,29</sup>. Thus it may be possible that GABA<sub>B</sub> receptors participate in the delayed suppression mechanism for directional selectivity, although only GABA<sub>A</sub> blockers have so far been reported to be effective in diminishing the directional selectivity<sup>30-33</sup>.

### Acknowledgements

We thank Drs. Robert Barlow, Erik Herzog, James Albert and Andy Ishida for comments on the manuscript. This study was supported partly by grants from Japan Science and Technology Corporation to H.U.

### References

1. Gibson JJ (1979), The Ecological Approach to Visual Perception. Houghton Mifflin, Boston
2. Marr D (1982), Vision: A Computational Investigation into the Human Representation and Processing of Visual Information. W.H. Freeman and Company, New York
3. Frost BJ, Nakayama K (1983), Single visual neurons code opposing motion independent of direction. *Science* 220: 744-745
4. Frost BJ, Sun H (1997), Visual motion processing for figure/ground segregation, collision avoidance, and optic flow analysis in the pigeon. In: Srinivasan

- V, Venkatesh S (eds), From living eyes to seeing machines. Oxford Univ. Press, Oxford, pp. 80-103.
5. Maturana HR, Frenk S (1963), Directional movement and horizontal edge detectors in the pigeon retina. *Science* 142: 977-979
6. Barlow HB, Hills RM, Levick WR (1964), Retinal ganglion cells responding selectively to direction and speed of image motion in the rabbit. *J Physiol* 173: 377-407
7. Michael CR (1968), Receptive fields of single optic nerve fibers in a mammal with an all-cone retina. II. Directionally selective units. *J Neurophysiol* 31: 257-267
8. Lipetz LE, Hill RM (1970), Discrimination characteristics of the turtle's retinal ganglion cells. *Experimentia* 26: 373-374
9. Daw NW, Beauchamp RD (1972), Unusual units in the goldfish optic nerve. *Vision Res* 12: 1849-1855
10. Stone J, Fukuda Y (1974), Properties of cat retinal ganglion cells: a comparison of W-cells with X- and Y-cells. *J Neurophysiol* 37: 722-748
11. Uchiyama H, Barlow RB (1994), Centrifugal inputs enhance responses of retinal ganglion cells in the Japanese quail without changing their spatial coding properties. *Vision Res* 34: 2189-2194
12. Barlow HB, Levick WR (1965), The mechanisms of directionally selective units in rabbit's retina. *J Physiol* 178: 477-504
13. Koch C, Torre V, Poggio T (1986), Computations in the vertebrate retina: Motion discrimination, gain enhancement and differentiation. *Trends Neurosci* 9: 204-211
14. Hildreth E, Koch C (1987), The analysis of visual motion: From computational theory to neuronal mechanisms. *Ann Rev Neurosci* 10: 477-533
15. Oyster CW (1990), Neural interactions underlying direction-selectivity in the rabbit retina. In: Blackmore C (ed) *Vision: Coding and Efficiency*. Cambridge Univ. Press, Cambridge, pp. 92-102.
16. Barlow H (1996), Intraneuronal information processing, directional selectivity and memory for spatio-temporal sequences. *Network: Computation in Neural Systems* 7: 251-259
17. He S, Masland RH (1997), Retinal direction selectivity after targeted laser ablation of starburst amacrine cells. *Nature* 389: 378-382
18. Masland RH, Mills JW, Cassidy C (1984), The functions of acetylcholine in the rabbit retina. *Proc R Soc Lond [Biol]* 223: 121-139
19. Keys RG (1981), Cubic convolution interpolation for digital image processing. *IEEE Trans. on Acoustics, Speech, and Signal Processing* 29: 1153-1160
20. Jones JP, Palmer LA (1987), The two dimensional spatial structure of simple receptive fields in cat striate cortex. *J Neurophysiol* 58: 1187-1211
21. Shapley R, Reid RC, Soodak R (1991), Spatiotemporal receptive fields and direction selectivity. In: Landy MS, Movshon JA (eds), *Computational Models of Visual Processing*. The MIT Press, Cambridge, Massachusetts, pp. 109-118.
22. DeAngelis GC, Ohzawa I, Freeman RD (1995), Receptive field dynamics in the central visual pathways. *Trends Neurosci* 18: 451-458
23. Wyatt H, Daw N (1975), Directionally sensitive ganglion cells in the rabbit retina: Specificity for stimulus direction, size, and speed. *J Neurophysiol* 38: 613-626
24. Amthor FR, Grzywacz NM (1993), Inhibition in ON-OFF directionally selective ganglion cells of the rabbit retina. *J Neurophysiol* 69: 2174-2187
25. Grzywacz NM, Amthor FR (1993), Facilitation in ON-OFF directionally selective ganglion cells of the rabbit retina. *J Neurophysiol* 69: 2188-2199
26. Yang G, Masland RH (1994), Receptive fields and dendritic structure of directionally selective retinal ganglion cells. *J Neurosci* 14: 5267-5280
27. Pan ZH, Slaughter MM (1991), Control of retinal information coding by GABAB receptors. *J Neurosci* 11: 1810-1821
28. McCormick DA (1990), Membrane properties and neurotransmitter actions. In: Shepherd GM (ed) *The Synaptic Organization of the Brain*, 3rd ed. Oxford Univ. Press, New York, NY, pp. .
29. Mody I, De Koninck Y, Otis TS, Soltesz I (1994), Bridging the cleft at GABA synapses in the brain. *Trends Neurosci* 17: 517-525
30. Wyatt HJ, Daw NW (1976), Specific effect of neurotransmitter antagonists on ganglion cells in rabbit retina. *Science* 191: 204-205
31. Caldwell JH, Daw NW, Wyatt HJ (1978), Effects of picrotoxin and strychnine on rabbit retinal ganglion cells: lateral interactions for cells with more complex receptive fields. *J Physiol Lond* 276: 277-298
32. Ariel M, Daw NW (1982), Pharmacological analysis of directionally sensitive rabbit retinal ganglion cells. *J Physiol (Lond)* 324: 161-185
33. Massey SC, Linn DM, Kittila CA, Mirza W (1997), Contributions of GABAA receptors and GABAC receptors to acetylcholine release and directional selectivity in the rabbit retina. *Vis Neurosci* 14: 939-948



# Hysteresis Phenomena in Depth Perception of a Moving Object

*Department of Computer Science and Systems Engineering  
Faculty of Engineering, Miyazaki University  
1-1, Gakuen Kibanadai Nishi, Miyazaki, 889-2192 Japan  
Hiroyasu Jinnai, Tetsuro Kitazoe, Tomohiro Shii*

## Abstract

Hysteresis phenomena are discussed in the three layered neural network model(TLNN) which are able to recognize temporally changing sequential stereovision. To test what TLNN predicts, we prepare eleven kinds of random dot stereograms including gray in addition to white and black. A subject sees stereograms one by one on the computer display. After the subject recognizes the central region deeper than the background, he or she will look the region in front of the back ground eventually. Similarly the reversed process is tested. The results of every subject show hysteresis phenomena clearly and is consistent with that predicted by TLNN.

## 1. Introduction

It seems that the hysteresis phenomena are seen often in a variety of perceptions in our brain. When input stimuli change gradually, the present perception is influenced by that caused by the former stimuli. This fact suggests that our perception is not invariant under time reversal. We will have different recognition between gradual input change from a state A to B and the reversed change from B to A. We study if the hysteresis phenomena occur in the sequential stereoscopic visual depth perception. We test the time reversibility of the processes when a depth of an object changes gradually to another depth[1].

Various models have been proposed to recognize stereoscopic vision for two 2-dimensional input maps entering into left and right eyes[[2][3][4][5][6][7][8]]. The famous neural network model for stereo vision was studied by Amari and Arbib[2]. In 1995, Reinman and Harken proposed a new equation with co-operation and competition which successfully recognized static stereovision[7], though their model did not have an obvious correspondence to the neural networks in the brain. When hysteresis phenomena are to be discussed, it is needed to construct a neural network model which enables to recognize temporal sequential stereo vision. It is, however, not easy to construct such a model, because winner and loser neurons must change their roles according to temporarily changing similarity maps between left and right eyes. When the sequential stereovision accompanies the depth change of an object, the process means in terms of binocular neurons activities that a winner neuron which corresponds to a specific depth of an object becomes a loser, handing over its role to a new winner neuron which corresponds to another depth. The primitive competition model by Amari and Arbib has a difficulty in the handing over process;

a winner cannot be a loser easily even if the input becomes weak[2]. It was also pointed out that Reinman-Harken model had serious difficulties in the recognition of moving objects[9]. Recently we presented a new model which was composed of three layered neural networks[10]. The model is able to recognize sequential stereo vision such as depth changing objects. It was also suggested that the model had hysteresis feature for the sequential depth perception[11].

In the present study we examine how the proposed model has hysteresis phenomena by using computer simulation for three dimensional objects. The test is carried out by making two 2-dimensional random dot stereograms for left and right eyes, where a squared region at the center of the random dot map for right eye is shifted to left from the corresponding part for left eye(state A). Thus, we will see the central squared region in front of the back ground after two pictures are fused. Next we make another pair of maps where central squared region for right eye is shifted to right compared to left eye map(state B). Here we see the squared region behind the back ground in this case. We introduce eleven stages changing gradually between A and B, where we use nine grades of gray dot by mixing A and B with different weight. The neural network equation processed the eleven kinds of input data for left and right eyes. After solving the equation, we got the neuron activities which gave different values for the path from A to B and the reversed one from B to A, showing hysteresis phenomena clearly.

We examine how the hysteresis result obtained from the three layered neural net equation will be recognized by human beings. We asked several subjects to see eleven kinds of maps on a computer display sequentially. The results were that all the subjects showed hysteresis phenomena, though the transition points from A to B or from B to A had different values among subjects.

## 2. Similarity Map in sequential Stereovision.

To process stereo vision in neural network model, we need to introduce the similarity function  $S(u, v, a, b)$  between the feature  $L$  of an image point  $(u, v)$  in the left image and  $R$  of  $(u + a, v + b)$  in the right image which is given as follows:

$$S(u, v, a, b) = E - \frac{\iint |L(x, y, u, v) - R(x + a, y + b, u + a, v + b)| dx dy}{\iint f(x, y) dx dy} \quad (1)$$

.where

$$L(x, y, u, v) = f(x - u, y - v)f_L(x, y) - \bar{g}_L(u, v) \quad (2)$$

$R(x, y, u, v)$  is defined correspondingly.  $E$  is a constant.  $g_L(x, y)$  and  $g_R(x, y)$  are the left and right stereo input data at the point  $(x, y)$ . The area which defines the features is given by the window function  $f(x, y)$ .  $\bar{g}_L(u, v)$  and  $\bar{g}_R(u, v)$  are the local mean gray values of the features corresponding to the image points  $(u, v)$ . Since the disparity can vary only between certain limits, the search has to be performed in between. In the definition of  $S(u, v, a, b)$ ,  $a, b$  are limited in the disparity search area(DSA) which is defined as  $-a_s \leq a \leq a_s, -b_s \leq b \leq b_s$ . We define a value of  $\lambda_{ab}^{uv}$  [7] as a deviation of the similarity measure from its mean value in the DSA and use  $\lambda_{ab}^{uv}$  as an input to the neural net equation which we will discuss in the followings.

Let us first study, for the shake of simplicity, one-dimensional stereo vision. Accordingly,  $\lambda_{ab}^{uv}$  was simplified as  $\lambda_a^u$ . The dynamic input from the moving object to left and right eyes were simplified as two sequential input, the first and the second, where the second input was presented after the first stereoscopic depth perception had been achieved.

The random 50 dots written below were used as the first input.

Left(1st to 50th)

111000000010101101011011001111111110100100100101011

Right(1st to 50th)

111000000010101101011011001111111110100100100101011

In these dots, the displaced right dots were produced by planting the left dots from 21st to 30th in the right-dot-places from 23rd to 32nd in order.

As the second sequential input, the same random dots for left eye were used. The second displaced dots for right eye were produced by planting the left dots from 21st to 30th in the right-dot-places from 19th to 28th in order. Thus the random 50 dots for second stage are given as

Left(1st to 50th)

111000000010101101011011001111111110100100100101011

Right(2nd to 50th)

111000000010101101101100111111111011110100100100101011

Based on these setups we see a bar from 21st to 30th two dots deeper than the background (1st stage) and then the bar coming up two dots in front of the background (2nd stage). The input data  $\lambda$  are calculated by using parameters  $E = 4, a_s = 3$  for one dimensional case.

### 3. Three Layered Neural Nets

The proposed networks are given by

$$\begin{aligned} \alpha_{ab}^{uv} &= -\alpha_{ab}^{uv} + A\lambda_{ab}^{uv} \\ &- B \sum_{a', b' \neq a, b} g(\xi_{a'b'}^{uv}(t)) + D \sum_{u', v' \neq u, v} g(\xi_{ab}^{u'v'}(t)) \end{aligned} \quad (3)$$

$$\beta_{ab}^{uv} = -\beta_{ab}^{uv} + g(\alpha_{ab}^{uv}) + g(\xi_{ab}^{uv}) \quad (4)$$

$$\xi_{ab}^{uv} = -\xi_{ab}^{uv} + f(\beta_{ab}^{uv}) \quad (5)$$

which are composed of  $\alpha, \beta$  and  $\xi$  layers.  $\alpha, \beta$  and  $\xi$  have suffices  $(a, b), (u, v)$  corresponding to those of input similarities  $\lambda_{ab}^{uv}$ , where  $(a, b)$  moves in disparity search area DSA while  $(u, v)$  shows coordinates of each image points. In eq.(3) and (4)  $g(u)$  is given by

$$g(u) = u^+ = \begin{cases} u & (u > 0) \\ 0 & (u \leq 0) \end{cases} \quad (6)$$

which was introduced in the neural net works equation by Amari and Arbib[2]. In eq.(5)  $f(x)$  is a well known sigmoid function given as

$$f(x) = \frac{\tanh(w(x - h)) + 1}{2} \quad (7)$$

In Fig.1 the three layered network is shown, composed of  $\alpha, \beta$  and  $\xi$  layers. The  $\alpha$  layer receives input  $\lambda$  and the third term in  $\alpha$  layer shows a competition in DSA. The forth term in the  $\alpha$  layer represents a cooperation among neighboring neurons in cooperative area(CA) which is defined as  $u - l \leq u' \leq u + l, v - l \leq v' \leq v + l$  (We take  $l = 3$ ). Since A.B.D are positive constants,  $\alpha$  has inhibitory coupling with  $\xi$  neuron in DSA and excitatory coupling with  $\xi$  neurons in CA. The  $\beta$  layer receives  $\alpha$  as input and also gets a feedback from  $\xi$  layer. Finally  $\xi$  layer processes the result of  $\beta$  layer by a sigmoid function.

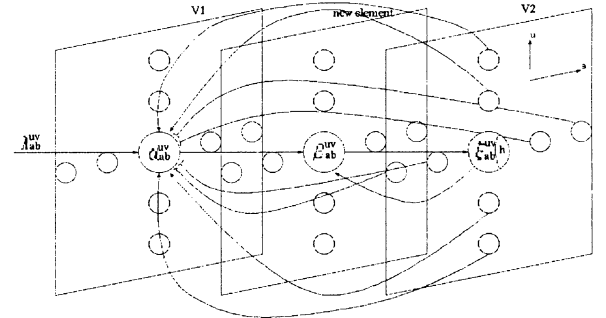


Figure 1: Three layered structure for eqs. (3),(4) and (5)

### 4. Stereovision and Hysteresis

To understand the qualitative feature of the equations, consider equilibrium solution  $\alpha_{ab}^{uv} = \beta_{ab}^{uv} = \xi_{ab}^{uv} = 0$ . Eqs.(3),(4),(5) are written as

$$\xi_{ab}^{uv} = f(g(\xi_{ab}^{uv}) + g(\alpha_{ab}^{uv})) \quad (8)$$

In Fig.2, the curves of  $y = x$  and  $y = f(g(x) + f(\alpha))$  are shown for changing value of  $\alpha_{ab}^{uv}$  from positive large one(a) down to small positive value (e). The solutions are given by

the intersection of the two curves, while the intermediate solution in (c) is not stable, because we take  $w = 2.5$ . If  $\alpha_{ab}^{uv}$  decreases from (a) to (e), the solution maintains  $\xi \approx 1$  until it reaches to the value of (d). On the contrary, if  $\alpha_{ab}^{uv}$  increases from (e) to (a), the  $\xi \approx 0$  solution is maintained until it reaches to the value of (b). From this fact we obtain two conclusions.

- (1) For large  $\alpha$ ,  $\xi$  has high value ( $\approx 1$ ) while it has low value ( $\approx 0$ ) for small  $\alpha$ .
- (2) The solution  $\xi$  has different path according to whether  $\alpha$  is increased or decreased.

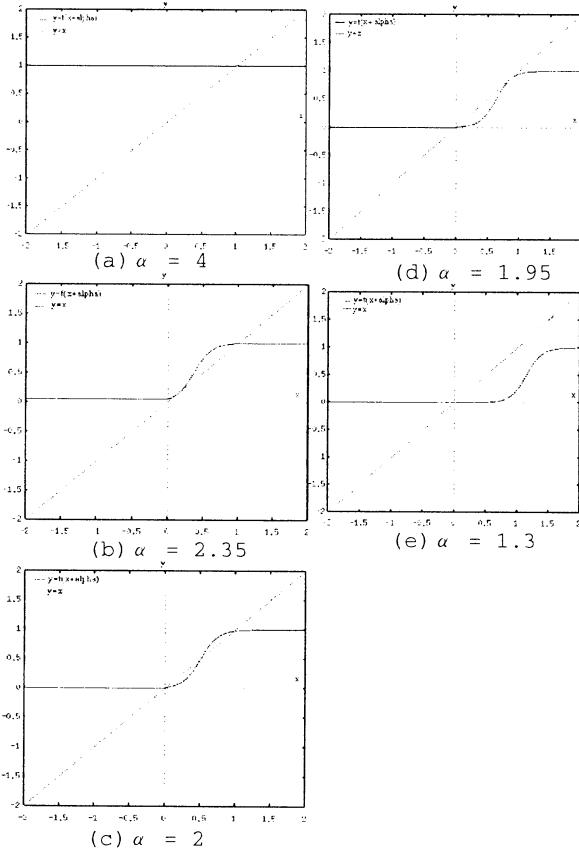


Figure 2: The curves of  $y = x$  and  $y = f(g(\xi) + g(\alpha))$

Since  $\alpha$  is influenced additively by the input  $\lambda$  and competition-cooperation terms accelerate winner-take-all (WTA) processes, the first conclusion leads to a stereo vision and the second one leads to hysteresis phenomena. In fact in Fig.3(a) we had a nice depth perception. Moreover, in Fig.3(b) we succeeded a depth perception ( $a = -2$ ) sequentially changed from depth ( $a = 2$ )

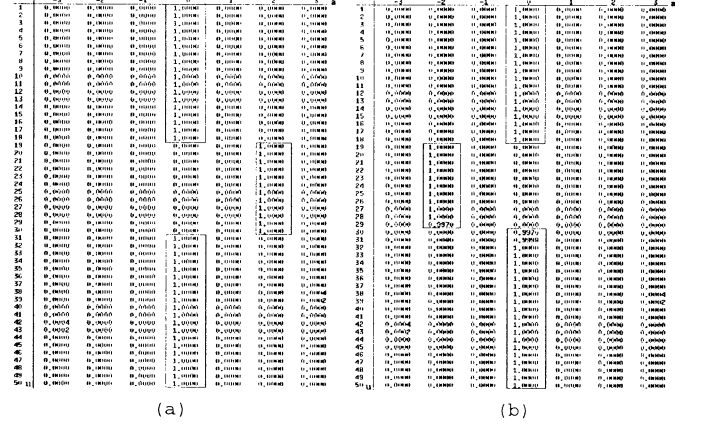


Figure 3: The results of simulation for depth  $a = 2$  (a) and then  $a = -2$  (b) with  $A = 25, B = 10, D = 0.5, w = 2.5, h = 2.5$  in eqs.(3),(4),(5) and (7)

## 5. Experimental Evidence for Hysteresis

To observe the hysteresis phenomena in the neural network equations, we perform a simulation by changing input from depth 2 bitmap  $\lambda_{ab}^{uv}(2)$  to depth -2 bitmap  $\lambda_{ab}^{uv}(-2)$  gradually and then carry out the process reversed. We introduce  $\lambda_{ab}^{uv}(\varepsilon)$  as

$$\lambda_{ab}^{uv}(\varepsilon) = (1 - \varepsilon)\lambda_{ab}^{uv}(2) + \varepsilon\lambda_{ab}^{uv}(-2) \quad (9)$$

where  $0 \leq \varepsilon \leq 1$  is considered as gray value ranging between  $a = 2$  and  $a = -2$  random dot stereograms. We obtained stable solutions  $\xi^u$  at each  $\varepsilon$  value.  $\varepsilon$  is increased from 0 to 1 and then decreased back to 0. Fig.4 shows the hysteresis phenomenon that  $\xi_{a=-2}^{u=25}$  which was 0 at the beginning increased suddenly to 1 at  $\lambda = 0.38$  when  $\lambda$  was increased and that it did not fall to zero until  $\lambda = 0.05$  when  $\lambda$  was decreased.

To test whether we actually experience the hysteresis phenomena, we made two dimensional random dot bit maps (100x100) for left and right eyes. First we made two kinds of bit maps R(2) and R(-2) for right eye. R(2)(R(-2)) is made by shifting two dots right(left) from left bit map L at the central squared region (60x60). If you see L and R(2)(R(-2)), you see the squared region two dots deeper(nearer) than the background by fusing the images. Now we make eleven kinds of right bit maps R( $\varepsilon$ ) on the X-window by introducing nine kinds of gray between white and black:

$$R(\varepsilon) = (1 - \varepsilon)R(2) + \varepsilon R(-2) \quad (10)$$

We employed 5 students to test the hysteresis. They looked at L and R( $\varepsilon$ ) and told us if they saw the stereo vision or not. We tested hysteresis by increasing  $\varepsilon$  from 0 to 1 by a step 0.1 and then decreasing  $\varepsilon$  from 1 to 0 in the same way. In Fig.5 typical hysteresis loop is shown.

In the figure, they saw the central squared region two dot deeper than the background until  $\varepsilon \leq \varepsilon_1$ . They did not see anything between  $\varepsilon_1 < \varepsilon \leq \varepsilon_2$ . After  $\varepsilon_2$ , they saw the central region two dots nearer than the background.  $\varepsilon_3$  and  $\varepsilon_4$  are

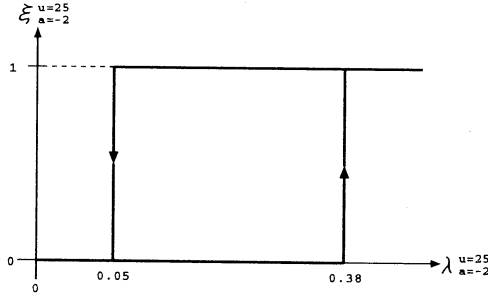


Figure 4: The solution of  $\xi$  has a different path according to whether  $\lambda$  is increased or decreased.

similarly defined when  $\varepsilon$  is decreased from 1 to 0.

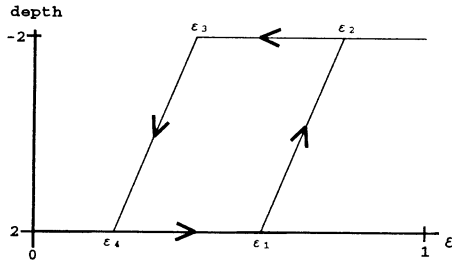


Figure 5: Typical example for hysteresis loop.

In the Table.1. results of  $\varepsilon$  are shown for each subject. We see all the subjects show hysteresis phenomena because  $\varepsilon_4 \leq \varepsilon_1, \varepsilon_3 \leq \varepsilon_2$ . It is also noticed that hysteresis phenomena differs from person to person judging from observed value of  $\varepsilon_i$ . Though  $S_5$  gave very extreme values for  $\varepsilon_i$ , we notice that the results are very much resemble to those of computer simulation given in Fig.4.

NAME	$\varepsilon_1$	$\varepsilon_2$	$\varepsilon_3$	$\varepsilon_4$
$S_1$	0.58(±.03)	0.50(±.00)	0.40(±.03)	0.46(±.00)
$S_2$	0.44(±.03)	0.77(±.03)	0.59(±.03)	0.37(±.02)
$S_3$	0.57(±.02)	0.64(±.04)	0.58(±.03)	0.49(±.02)
$S_4$	0.62(±.02)	0.70(±.02)	0.40(±.02)	0.31(±.02)
$S_5$	0.88(±.03)	1.00(±.03)	0.14(±.03)	0.00(±.02)

Table 1: Hysteresis results for  $S_1 - S_5$ .

## 6. Conclusion

The three layered neural network model proposed recently is the one which enabled to recognize sequential stereovision at the first time. We have tested hysteresis which the model predicts. The test is of new kind where depth 2 bit maps change to depth -2 bit maps gradually. The results of every subject

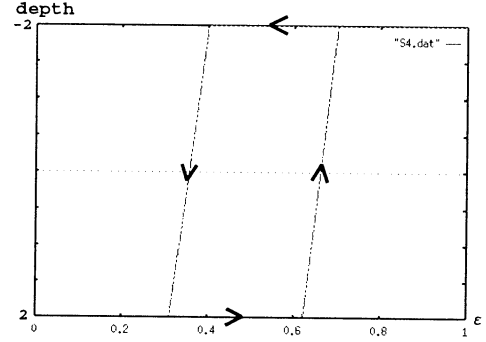


Figure 6: The typical results( $S_4$ ) for hysteresis experiment.

showed hysteresis clearly, which was consistent with the one predicted by the model.

## References

- [1] Fender D and Julesz B (1967)  
Extension of Panum's fusional area in binocularly stabilized vision. J.Opt.Soc.Am. 57:819-830
- [2] Amari,S and Arbib, M A (1977)  
Competition and Cooperation in Neural Nets. Systems Neuroscience:119-165
- [3] Geman S and Geman D (1984)  
Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. IEEE Trans. Pattern Anal. Machine Intell. PAMI-6(6):721-741
- [4] Terzopoulos D, Witkin A and Kass M (1987)  
Stereo Matching as Constrained Optimization Using Scale Continuation Methods. Optical and Digital Pattern Recognition. SPIE 754:92-99
- [5] Hentschel H G E and Fine A (1989)  
Statistical Mechanics of Stereoscopic Vision. Phys. Rev. A 40:3983-3997
- [6] Yuille A, Geiger D and Bulthoff H H (1991)  
Stereo Integration, Mean Field Theory and Psychophysics. Network 2:423-442
- [7] Reimann D and Haken H (1994)  
Stereo Vision by Self-Organization. Biol. Cybern. 71:17-26
- [8] Yoshitomi Y, Kanda T, Kitazoe T and Shii T (1998)  
Neural Nets Pattern Recognition Equation for Stereoscopic Vision. Trans.IPSJ 39 1:29-38
- [9] Yoshitomi Y, Kitazoe T, Tomiyama J and Tatebe Y (1998)  
Sequential Stereo Vision and Phase Transition. Proc. of Third Int. Symp. on Artificial Life and Robotics:318-323
- [10] Kitazoe T, Tomiyama J, Yoshitomi Y, and Shii T (1998)  
Sequential Stereoscopic Vision and Hysteresis. Proc. of Fifth Int. Conf. on Neural Information Processing. 391-396
- [11] Haken H (1991)  
Synergetic Computers and Cognition. Springer, Berlin Heidelberg, New York

## Visual Perception Depends on Auditory Stimuli?

Kazuchika Manabe and Hiroshi Riquimaroux

Department of Knowledge Engineering and Computer Science, Faculty of Engineering, Doshisha University,  
Kyotanabe-City, Kyoto Japan 610-0321 (E-mail) Kmanabe@mail.doshisha.ac.jp

### Abstracts

Many studies have reported that visual information alters auditory perception when stimuli in both modalities are presented at a time, such as McGurk- and ventriloquism-effects. However, little has been shown that auditory stimulus changes visual perception. This asymmetry might have indicated that the visual modality overrules the auditory modality. Here, we report new evidence that an auditory input does alter visual perception. When two stationary lights, mounted a short distance apart, are alternately lit at a certain rate, we can see an apparent movement of the light from one to the other. We found that the velocity perception of this apparent motion was significantly altered by simultaneous presentation of sound between the offset of the first light and the onset of the second. The shorter the sound duration, the faster the perceived velocity was, even though the intervals between lights were physically constant. Our data have shown that there was no clear effect of sound intensity on the velocity estimation of visual apparent motion. The finding indicates that the sound duration appears to be a key factor to control the velocity estimation of the apparent motion.

**Key words:** cross-modal integration effect, velocity estimation, visual apparent motion, auditory

### Introduction

When stimuli in both visual- and auditory-modalities are presented at the same time, visual information usually alters auditory perception. When sound is presented from a certain position, for example,

we can locate the sound source at the position. On the other hand, we miss-locate the voice source at a position where a talking face without voice, like a doll of ventriloquism is presented even if the position of the actual voice source is different from the position ("ventriloquism-effects"<sup>1</sup>). We can correctly perceive /ga/ sound, if /ga/ sound is presented only by a speaker. However, we misperceive /ga/ sound as /da/ sound, if a face that is producing a /ba/-lip movement is presented with /ga/ sound (McGurk-effects<sup>2</sup>). Those are typical examples of "visual-dominance effects." However, little has been shown that auditory stimulus changes visual perception (but see McFarland & Kennison<sup>3</sup>; Morriongiello<sup>4</sup>; Regan & Spekreijse<sup>5</sup>; Staal & Donderi<sup>6</sup>; Stein et al.<sup>7,8,9</sup>). This asymmetry might have indicated that the visual modality overrules the auditory modality when visual and auditory stimuli are presented at the same time. On the contrary, we found new evidence that an auditory input does alter visual perception. We found that a velocity perception of visual apparent motion was significantly altered by simultaneous presentation of a sound.

### Experiment 1: Effects of Sound Duration

#### Method

*Subjects.* Eleven college students participated in the experiments.

*Materials and procedures.* Three red light-emitting-diodes (LEDs), 5 mm in diameter, were

mounted horizontally (left, center and right) on a black cardboard. Each LED was apart 10 mm from the center to center. The visual angle was  $1.9^\circ$  at subject's eye level. The luminance of each LED was  $306 \text{ cd/m}^2$ , measured by a spot photometer (BM-3, TOPCON). We asked subjects to compare the velocity of apparent motion from the left LED to the center LED, the standard (Speed (F) in Figure 1) with the center to the right, the comparison (Speed (R) in Figure 1). Subjects responded by clicking buttons indicating "Faster", "Same" and "Slower" according to their estimations. When subjects perceived the comparison was faster than the standard, subjects needed to click the "Faster" button, and so on. Each LED was illuminated for one second in a sequence, from the left to the right through the center. The interval between the offset of the left-LED and the onset of the center-LED (IOO) was the same as IOO between the center-LED and the right-LED. The IOO was set at either 56, 85 or 130 ms.. A brief wide-band noise burst was presented to the right ear by a headphone (ATH-F3, audio-technica®) and an A/D board (Sound Blaster 16, Creative Technology Ltd.) during the interval between the offset of the center-LED and the onset of the right-LED. Sound was started just after the offset of the center LED. Each sound had 10-ms rise and fall times. The sampling rate was 44.1 kHz and the resolution was 16

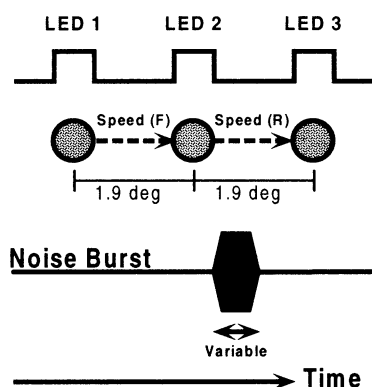


Figure 1. Configuration of visual and auditory stimuli in time and space in Experiment 1. Duration of wide-band noise burst was varied.

bits. Sound intensity was set at 89-dB SPL as measured by a sound level meter (Type 6030, ACO) with a dummy head (Type 4128, Brüel & Kjær). The sound duration was one second for the sound intensity measurement. In the experiment, the burst duration was randomly changed within a session from 20 to 56 ms in 4-ms steps for 56-ms IOO condition, from 40 to 85 ms in 5-ms steps for 85-ms IOO condition and 40 to 130 ms in 10-ms step for 130-ms condition. Therefore, noise bursts terminated before the onset of the right-LED except for the longest noise burst (see Figure 1). Each subject had read an instruction that the sound duration was irrelevant to the velocity of apparent motion before the test sessions (see Appendix).

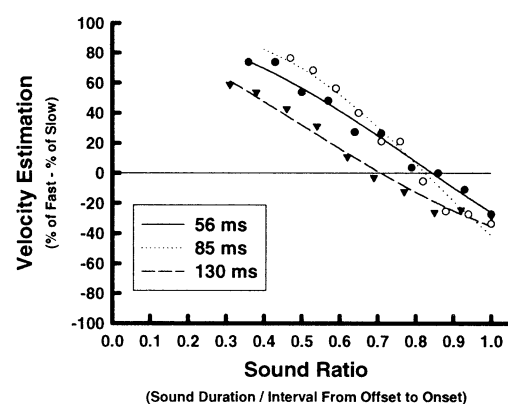


Figure 2. Velocity estimation subtracting percentage of "Slower" estimation from percentage of "Faster" estimation when sound duration was varied within a session. Sound ratio was calculated by dividing the sound duration by the entire interval between the offset and the onset of adjacent LEDs (IOO). A horizontal line indicates the "same" judgement. Positive values show that subjects perceived the comparison velocity with sound faster than the standard velocity without sound. Negative values indicate the opposite. Each curve was estimated by a cumulative-normal-distribution model. A solid line represents a condition in which IOO was 56-ms, a dotted line 85-ms IOO condition and a dashed line 130-ms IOO condition.

## Results

Figure 2 shows averaged velocity estimation

across different sound duration taken from 11 subjects. The velocity estimation was clearly affected by the sound duration in each condition ( $p < 0.0001$ ). When the sound duration was shorter than 84 % of IOO (47 ms) in 56-ms IOO condition, subjects perceived that a motion from the center to the right was faster than that from the left to the center. On the other hand, when the sound duration was longer than 84 % of IOO in 56-ms IOO condition, subjects perceived that a motion from the center to the right was slower than that from the left to the center. The critical sound duration was 82 % (70 ms) in 85-ms IOO condition and 70 % (91 ms) in 130-ms IOO condition.

### Experiment 2: Effects of Sound Intensity

Loudness is perceived differently depending upon the sound duration, which is less than 200 ms. The shorter the sound is, the lower the perceived loudness (see Gulick<sup>10</sup>). Because we used the sound duration shorter than 200 ms, effects of loudness on the velocity estimation were examined.

### Method

**Subjects.** The same 11 College students participated in the experiments.

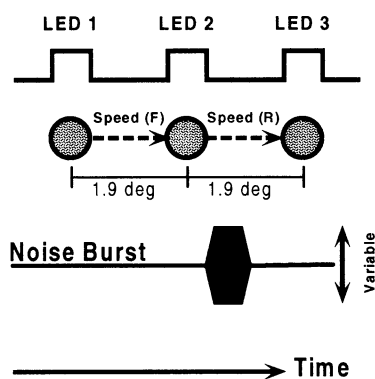


Figure 3. Configuration of visual and auditory stimuli in time and space in Experiment 2. Intensity of wide-band noise burst was varied.

**Procedures.** Relative sound intensity was varied from the original sound intensity (89 dB SPL) that was used in the first experiment to -40 dB (59 dB SPL) in 5-dB steps with the sound duration fixed at 85 ms (see Figure 3). Subjects judged the velocity under three different sound duration, 45 ms, 63 ms and 80ms. Each subject had read an instruction that loudness was irrelevant to the velocity of apparent motion before the test sessions.

### Results

Figure 4 shows velocity estimation across different sound intensities. As intensity of sound increased, subjects perceived the velocity of comparison (Speed (F) in Figure 3) was somewhat faster than the velocity of standard (Speed (R) in Figure 3). However, the effect was not statistically significant in the present experiment in each condition ( $p > 0.05$ ).

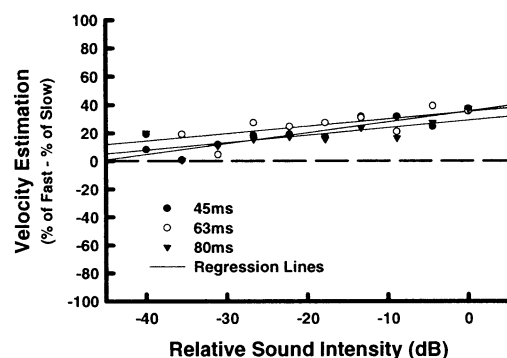


Figure 4. Velocity estimation subtracting percentage of "Slower" estimation from percentage of "Faster" estimation when sound intensity was varied within a session. A dashed line indicates the "same" judgement. Positive values indicate that subjects perceived the comparison velocity with sound was faster than the standard without sound. Negative values show the opposite. Closed circles represent 45-ms sound duration condition, open circles 63-ms sound duration condition and closed triangles 80-ms condition. Regression lines were calculated by the least square method. The  $r^2$  of 45-ms condition was 0.82, the  $r^2$  of 63-ms condition 0.52 and the  $r^2$  of 130-ms condition 0.54, respectively.

## General Discussion

In the first experiment, the shorter sound that would be perceived less loud than the longer sound produced the faster velocity perception. On the contrary, less loud sound produced somewhat slower velocity perception in the second experiment. Therefore, the altered velocity perception in the first experiment can not be attributed to the difference in perceived loudness of sound caused by the difference in sound duration. The results indicated that sound duration appears to be a key factor in the velocity estimation of visual apparent motion in the present situation. The most effective sound that can alter velocity perception of visual apparent motion may be a short and loud sound.

At this time, the mechanism underlying the perceived alteration in velocity stays unclear. This cross-modal integration effect, however, may be attributed to everyday experience in which a moving object in high velocity makes shorter sound than a low-speed object, like racing cars. Additional experiments are necessary to uncover the mechanism.

## Conclusion

A velocity perception of visual apparent motion was significantly altered by simultaneous presentation of sound. The shorter the sound duration, the faster the perceived velocity was, even though the intervals between lights were physically constant. Loudness of sound also may play a minor role in alteration in velocity.

## Acknowledgements

This study was preformed through Special Coordination Funds from the Science and Technology Agency of the Japanese Government.

## References

1. McGurk H, MacDonald J (1976), Hearing lips and seeing voices. *Nature* 264: 746-748.

2. Jack CE, Thurlow WR (1973), Effects of degree of visual association and angle of displacement on the "ventriloquism" effect. *Percept Motor Skills* 37: 967-979.
3. McFarland RA, Kennison RF (1988), Asymmetrical effects of music upon spatial-sequential learning. *J Gener Psych*, 115: 263-272.
4. Morrongiello BA (1988), Infants' localization of sounds along the horizontal axis: Estimates of minimal audible angle. *Develop Psych*, 24: 8-13.
5. Regan D, Spekreijse, H (1977), Auditory-visual interactions and correspondence between perceived auditory space and perceived visual space. *Percept*, 6: 133-138.
6. Staal HE, Donderi DC (1983), The effect of sound on visual apparent movement. *Amer J Psych*, 96: 95-105.
7. Stein BE, London N, Wilkinson LK, Price DD (1996), Enhancement of perceived visual intensity by auditory stimuli: A psychophysical analysis. *J Cognit Neurosci*, 8: 497-506.
8. Stein BE, Meredith MA (1993), *The Merging of the Senses*. Cambridge, MA: Bradford/MIT Press
9. Stein BE, Meredith MA, Huneycutt WS, McDadeL (1989), Behavioral indices of multisensory integration: Orientation to visual cues is affected by auditory stimuli. *J Cognit Neurosci*, 1: 12-24.
10. Gulick WL (1971), *Hearing: Physiology and psychophysics*. New York: Oxford University Press; pp. 148.

## Appendix

*Instruction in Experiment 1:* The present task is to estimate velocity of movement of a red light. At first, the left light will be lit and turned off. Just after the offset of the left light, the center light will be lit. At the time, you may be able to see a movement of light from the left to the center. In the next, the center light will be turned off and the right light will be turned on. At this time, you may be able to see a movement of a light from the center to the right. Your task is to compare the velocity of the movement of a light from the left to the center to the velocity of the movement of a light from the center to the right. If the velocity of a movement from the center to the right is faster than the velocity of a movement from the left to the center, click a button indicating "Faster". If the velocity of a movement from the center to the right is slower than the velocity of a movement from the left to the center, click a button indicating "Slower". If the velocity of a movement from the center to the right is the same as the velocity of a movement from the left to the center, click a button indicating "Same". A sound will be presented during a motion. The sound duration is irrelevant to the velocity of a light. However, please listen to the sound and estimate the velocity of a light. Please wear a headphone and click "OK" button, if you are ready.



# Speech Recognition using Stereovision Neural Network Model

*Department of Computer Science and Systems Engineering*

*Faculty of Engineering . Miyazaki University*

*1-1 . Gakuen Kibanadai Nishi . Miyazaki . 889-2192 Japan*

*Tetsuro Kitazoe. Sung-Il Kim. Tomoyuki Ichiki*

## ABSTRACT

The stereo vision neural-net equation, known to process a depth perception, is applied to speech recognition. We use a recently developed three layered neural net (TLNN) equation with competition and cooperation for stereo vision. We use a Gaussian PDF to represent memorized data of each phoneme in our memory, and the similarities of an input phoneme with respect to the memorized ones were calculated. The TLNN is applied to the similarities with best 5 hypotheses among 24 kinds of phonemes. The average rates for speaker independent recognition are 78.05 % for 216 word database and 78.94 % for 240 word database by TLNN equation which are compared to 71.56 % and 72.37 % by HMMs, respectively.

## 1. INTRODUCTION

Recently, many studies have been conducted for improvement of large vocabulary continuous speech recognition. There are two main trends in the art. One is to develop good acoustic models, such as triphone models with mixed Gaussian distribution and/or with tree based clustering models. The other is to reduce perplexity by using language models, such as n-gram and context free grammar. It was reported recently[1] that improving the acoustic models is much more effective than reducing perplexity; the improvement in word recognition rate achieved by increasing phoneme recognition by merely 1-2 % corresponds to the improvement achieved by decreasing perplexity as much as 10 - 20 %.

We applied stereo vision neural-net equations, known to process a depth perception, to speech recognition. In the stereo vision, the two-dimensional images of a 3-dimensional object are captured at two different points, left and right eyes. The local disparity (or similarity) between the two 2-dimensional images are determined through neural-net equations with competition and cooperation, whereby a clear depth perception can be obtained[2,3,4]. In real life speech recognition, it is considered that characteristic features of each phoneme are stored in our memory through daily life training and input speech data are compared with the memorized data to estimate their similarity (or disparity) for each phoneme.

We use a recently developed three layered neural net (TLNN) equation for stereo vision[4] to determine similarities among phonemes. When the equation is applied to phoneme recognition, it develops competition among similarities of different phonemes and cooperation among neighboring frame data, and a so-called \*winner-take-all process\* selects a specific phoneme as a recognized one, beating others down to zero

. We used a Gaussian PDF (Probability Density Function) to represent memorized data of each phoneme in our brain, and the similarities of an input phoneme with respect to the memorized ones were calculated. The simulation was performed for Japanese phoneme database[5,6]. It was found that the neural-net equation gave a clear recognition to each phoneme.

As training data, we used 4000 words recorded by 10 different speakers and 500 sentences recorded by 6 different speakers. We run recognition tests by using 10 dimensional MFCC coefficients and their derivatives. The recognition rates were greatly raised when each phoneme was divided into two parts, before and after the mid frame position, and the respective similarities were calculated separately. The TLNN is applied to the similarities with best 5 hypotheses among 24 kinds of phonemes. The average rates for speaker independent recognition were 78.05 % for 216 word database and 78.94 % for 240 word database by TLNN equation which were compared to 71.56 % and 72.37 % by Hidden Markov Model(HMM), respectively.

## 2. SIMILARITY MEASURE AND STEREO VISION NEURAL NET EQUATION

The speech(phoneme) recognition system by TLNN equation is divided into three main processes:

- (1) A number of training speech data are stored and classified for each phoneme. The data are supposed to be memorized in the brain with a standard form such as Gaussian PDF of cepstrums for each phoneme.
- (2) An input phonemes are referred to these memorized phoneme data and a similarity measure is obtained by comparing the input phoneme data with the memorized PDF of each phoneme.
- (3) Suppose that there is a neuron activity  $\xi_u^a$  in accordance with the similarity measure  $\lambda_u^a$  to a certain phoneme /a/ at the frame member u.

The TLNN equation processes the activity  $\xi_u^a$  to move toward a stable point after the equation receives the similarity measure as an input and a recognition is achieved when it reaches to a stable state.

The memorized data are expressed in terms of Gaussian PDF for input o.

$$N(o; \mu_a, \Sigma_a) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_a|}} e^{-\frac{1}{2}(o-\mu_a)^t \Sigma_a^{-1} (o-\mu_a)} \quad (1)$$

where  $\mu_a$  is a mean value of training cepstrum data of a phoneme /a/.  $\Sigma_a$  is given by

$$\Sigma_a = \frac{1}{N} \sum_{n=1}^N (o_n - \mu_a)(o_n - \mu_a)^t \quad (2)$$

where  $O_n$  is a training data of a phoneme /a/. The normalized similarity  $\lambda_u^a$  of input data  $O_u$  at u-th frame to a certain phoneme /a/ is defined as

$$\lambda_u^a = \frac{N(o_u; \mu_a, \Sigma_a) - \langle N \rangle}{\langle N \rangle} \quad (3)$$

where  $\langle N \rangle$  means an average over phonemes .

As a neural net equation we employ TLNN equation for stereo vision which is successful for recognizing stereo vision . The equation processes input similarity measures for two 2-dimensional data from left and right eyes and arrive at a definite depth perception through competition and cooperation processes . We use this equation for a speech recognition by using similarity measures  $\lambda_u^a$  between memorized PDF's and input phoneme data . The TLNN equation is given as

$$\dot{\xi}_u^a(t) = -\xi_u^a(t) + f(\beta_u^a) \quad (4)$$

where  $f(x)$  is a well known as sigmoid fuction given by

$$f(x) = \frac{\tanh(w(x-h)) + 1}{2} \quad (5)$$

$$\dot{\beta}_u^a = -\beta_u^a + g(\alpha_u^a) + g(\xi_u^a) \quad (6)$$

where  $g(u)$  is a function given by

$$g(u) = u^+ = \begin{cases} u, & u > 0 \\ 0, & u \leq 0 \end{cases} \quad (7)$$

$$\dot{\alpha}_u^a = -\alpha_u^a + A\lambda_u^a - B \sum_{a' \neq a} g(\xi_{u'}^{a'}(t)) + D \sum_{u'=n-l}^{u+l} g(\xi_{u'}^a(t)) \quad (8)$$

where A,B,D,w,h are positive definite constants . The neural network for these equations has a three layered structure as shown in figure 1 .

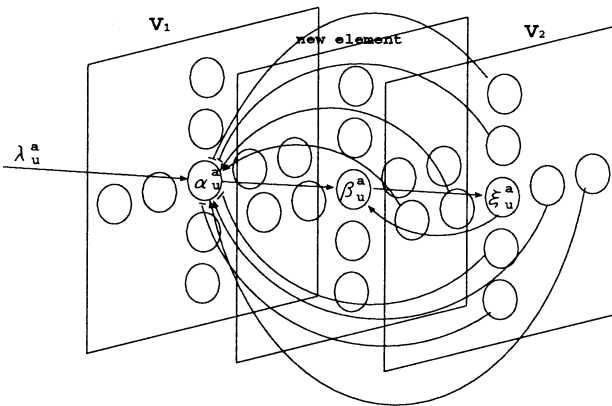


Figure 1: Three layered structure for TLNN equations

$\alpha_u^a$  receives not only similarity  $\lambda_u^a$  but also influence of neighboring activities  $\xi_{u'}^{a'}$  when  $\alpha_u^a$  is zero . The third term in  $\alpha_u^a$  represents a competition with activities  $\xi_u^a$  of other phonemes and the forth term does cooperation of the same phonemes at the neighboring frames  $u-l \leq u' \leq u+l$  . As stated later, however, they converged to the same values which are independent of the initial condition .

### 3. TLNN EQUATION WITH COOPERATION AND COMPETITION

Similarity measures  $\lambda$  are input to TLNN equation (4)(6)(8) only through  $\alpha$  and which plays an important role in the equations . To understand the qualitative feature of the equations, consider equilibrium solution  $\dot{\alpha}_u^a = \dot{\beta}_u^a = \dot{\xi}_u^a = 0$  . Equation (4)(5)(6) are written as

$$\xi_u^a = f(g(\alpha_u^a) + g(\xi_u^a)) \quad (9)$$

In figure 2, the curves of  $y = x$  and  $y = f(g(\alpha) + g(\xi))$  are shown for changing value of  $\alpha_u^a$  from positive large one (a) down to small positive value (d) .

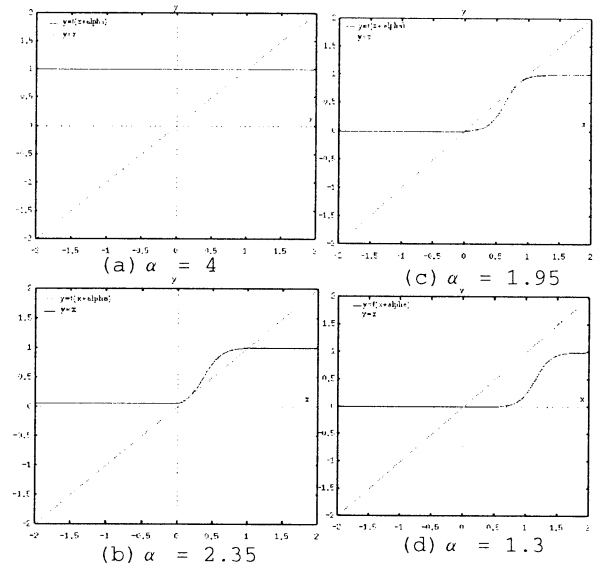


Figure 2: The curves of  $y = x$  and  $y = f(g(\alpha) + g(\xi))$

The solutions are given by the intersection of the two curves because we take  $w = 2.5$  . If  $\alpha_u^a$  decreases from (a) to (d), the solution maintains approximately  $\xi_u^a = 1$  until it reaches to the value of (d) . On the contrary, if  $\alpha_u^a$  increases from (d) to (a), approximately  $\xi_u^a = 0$  solution is maintained until it reaches to the value of (b) . From this fact we obtain two conclusions .

- (1) For large  $\alpha$ ,  $\xi$  has high value (approximately 1) while it has low value (approximately 0) for small  $\alpha$  .
- (2) The solution  $\xi$  has different path according to whether  $\alpha$  is increased or decreased .

Since  $\alpha$  is influenced additively by the input  $\lambda$  and the competition-cooperation terms accerelate winner-take-all processes, the first conclusion leads to a phoneme recognition and

the second one leads to hysteresis phenomena . We call a winner neuron when  $\xi$  gets a positive value finally and a loser neuron when it becomes zero, losing whole activity . In actual time dependence, the situation is more complicated because  $\alpha$  depends on neighboring  $\xi$ 's and thus varies with time .

$\mathbf{n}$	$\mathbf{m}$	$\mathbf{o}$	$\mathbf{g}$	$\mathbf{w}$
0.172669	0.007747	-0.179798	0.068170	-0.317374
0.047739	0.021944	0.012922	0.106335	-0.377080
-0.053358	-0.254189	0.174464	0.140137	-0.321096
-0.020677	-0.345811	0.166542	0.152011	-0.270617
0.071875	-0.109546	0.026478	0.047362	-0.181884
0.164128	-0.066376	-0.075502	0.000766	-0.187911
0.074848	0.021229	0.011177	-0.173780	-0.040727
0.075048	-0.128097	0.029788	-0.138120	0.028273
0.151001	-0.058196	-0.134549	-0.094952	-0.014505
0.181342	-0.005437	-0.214245	-0.072309	-0.070694
0.132347	0.084662	-0.163194	-0.224362	-0.046461
0.052027	0.157427	0.039553	-0.173396	-0.324618
0.112184	0.316814	0.044812	-0.315088	-0.632532
0.088750	0.277316	0.008593	-0.229108	-0.520211
0.064446	0.061100	0.028512	-0.384859	0.038372

Figure 3: An example of a similarity map

Figure 3 shows an example of a similarity map where the input phoneme is actually pronounced as /n/ and the phonemes /n/./m/./o/./g/./w/ with best five similarities are selected . Figure 4 shows results of recognition after TLNN equations are applied . /n/ is a winner for the frames 1-11, while /m/ is a winner for the frames 12-15 in this example .

$\mathbf{n}$	$\mathbf{m}$	$\mathbf{o}$	$\mathbf{g}$	$\mathbf{w}$
5.282858	-0.000000	-0.000000	0.000000	0.000000
5.044124	0.000000	0.000000	0.000000	0.000000
5.854819	0.000000	0.000376	0.000001	0.000000
5.706582	-0.000000	0.000000	0.000000	0.000000
5.582782	-0.000000	0.000000	0.000000	0.000000
5.479568	-0.000000	0.000000	0.000000	0.000000
5.218249	0.000000	-0.000000	0.000000	0.000000
4.369046	-0.000000	0.000000	0.000000	0.000000
4.728297	0.000000	-0.000000	-0.000000	-0.000000
4.477395	0.000000	0.000000	0.000000	0.000000
4.362889	0.000001	-0.000000	-0.000000	-0.000000
0.000005	3.567760	-0.000000	-0.000000	-0.000000
0.000000	3.731247	0.000000	-0.000000	-0.000000
-0.000000	3.905251	-0.000000	-0.000000	0.000000
-0.000000	4.125674	0.000000	-0.000000	-0.000000

Figure 4: Recognition results using TLNN equation with  $A=3.0$ ,  $B=3.5$ ,  $D=1.5$ ,  $w=2.5$ ,  $h=0.5$ ,  $l=4$

Thus, we conclude /n/ is correctly recognized in average . To get an understanding for the processes dynamically, we notice the sigmoid form which is written for the typical values of  $\alpha$ , with  $A=3.0$ ,  $B=3.5$ ,  $D=1.5$ ,  $w=2.5$ ,  $h=0.5$ ,  $l=4$  . The TLNN equation let  $\xi$ 's move to the  $\xi=0$  solution .

Since the stable solution for the equation is decided by the minima of the sigmoid form, the simulation shows that the solutions are uniquely determined independent of the initial values which are set all  $\xi$ 's=1 in the following discussions .

Figure 5,6 show time dependent behaviors for  $\alpha$ 's and  $\xi$ 's, at the 5-th frame for phoneme /n/./m/./o/./g/./w/ when the similarity map of figure 3 is input . Initially, only the difference among phonemes comes from  $\alpha$ 's in TLNN equation, where input  $\lambda$ 's are different . The whole  $\xi$ 's begin to decrease as shown in figure 6 due to the sigmoid form for  $\alpha < 0$  . Then,  $\alpha$  for /n/ with the biggest  $\lambda$  begins to take positive values as the competition term increases . When  $\alpha^n$  becomes positive, the activity  $\xi^n$  turns to increase according to the sigmoid form . Figure 6 shows  $\xi^n$  turns to increase . At this stage, it is noticed that the cooperation term in  $\alpha^n$  helps to rise  $\alpha^n$  and accelerate  $\xi^n$  in increasing .  $\alpha$ 's for other phonemes, on the other hand, begin to

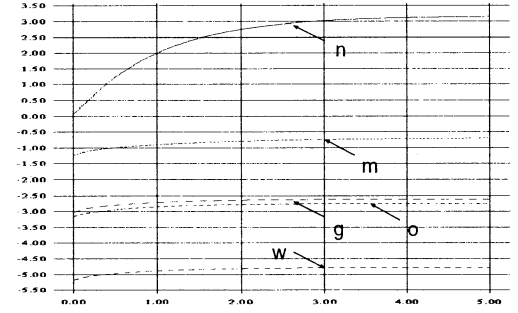


Figure 5: Time dependent behaviors for  $\alpha$

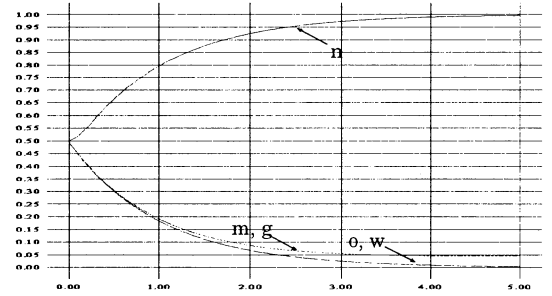


Figure 6: Time dependent behaviors for  $\xi$

decrease in figure 5 because of the increase of their competition terms due to increasing  $\xi^n$  . Therefore,  $\xi$ 's of other phonemes continue to decrease and finally goes to zero as shown in figure 6 . Thus, the phoneme /n/ is recognized through TLNN equations . In conclusion, cooperation and competition in TLNN equation play a good role to make a definite recognition as shown in figure 4 .

## 4. TECHNICAL IMPROVEMENT AND EXPERIMENTAL RESULTS

To make Gaussian PDF for each phoneme from training data, we extracted labeled phonemes from ATR data[6] composed of 4000 words spoken by 10 male speakers repeatedly and from ASJ data of 500 sentences[6] by 6 male speakers . The input data for recognition experiment were composed of two kinds, one from database of 216 words and the other from one of 240 words, spoken by 3 male speaker independent respectively . The speech data were analyzed as follows:

To compare our neuron model with the conventional model, the phoneme recognition experiment was performed for HMMs with single mixture and three states, by using exactly the same data as was used for the neural net equation . We took 24 simple labeled phonemes from ATR data for recognition .

We considered to improve our neuron model . If we looked at the phoneme data, it was noticed that there were considerable differences in the characteristic features of cepstrum data between those of former half part and latter half part of frame

Sampling rate	16KHz.16Bit
Pre-emphasis	0.97
Window	16 msec Hamming window
Frame period	5 ms
Feature parameters	10 order MFCC +10 order delta MFCC

Table 1: Analysis of speech signal

Kinds of data	Kinds of Model	Recognition rates
216 data	HMM	71.56
"	TLNN	78.05
240 data	HMM	72.37
"	TLNN	78.94

Table 2: The rates of speaker independent recognition depending on two kinds of data

Pho	216 data		240 data	
	HMM	TLNN	HMM	TLNN
NG	53.46(159)	83.54	59.62(156)	89.03
a	92.55(483)	94.62	93.85(390)	96.41
b	76.62(77)	77.22	86.79(53)	86.79
ch	84.62(65)	83.08	100.00(12)	83.33
d	69.84(63)	71.88	74.07(27)	62.96
e	64.77(264)	86.74	80.86(162)	96.30
g	57.14(77)	48.05	45.71(35)	38.89
h	63.46(52)	51.92	53.33(15)	60.00
i	69.16(308)	86.04	84.18(297)	96.97
j	97.01(67)	94.03	93.10(29)	93.10
k	55.25(219)	67.58	67.02(282)	58.16
m	61.90(105)	47.17	86.67(30)	60.00
n	44.30(79)	38.75	50.00(24)	50.00
o	70.58(486)	91.56	66.67(597)	89.45
p	64.00(25)	40.00	100.00(9)	77.78
r	62.34(77)	25.97	42.30(331)	35.65
s	89.01(91)	86.81	76.40(89)	78.65
sh	96.05(76)	84.21	91.11(45)	95.56
t	4.35(69)	28.99	15.38(39)	48.72
ts	65.22(23)	86.96	89.74(39)	89.74
u	94.78(115)	62.07	59.80(301)	68.00
w	84.38(32)	69.70	91.03(145)	74.15
y	61.36(44)	72.73	87.30(252)	92.86
z	87.76(49)	87.76	93.10(29)	93.33
ALL	71.56(3105)	78.05	72.37(3388)	78.94

% percentage (total number of phonemes)

Table 3: Recognition rates of test data

data. Therefore, we decided to divide cepstrum data into two parts, the former half and the latter half of phoneme data. We made two Gaussian PDF's for each phoneme separately. Input data which were similarly divided into two parts were compared to corresponding part of the Gaussian PDF's separately and a similarity map was obtained. After TLNN equation processed these similarity maps, the resulting recognition was shown in table 2 and 3, where we could see a remarkable improvement was obtained. Our neuron model gave approximately 7 % higher in average recognition rates than the performance of HMMs for 216 and 240 word databases.

## 5. CONCLUSION AND DISCUSSION

We applied recently developed TLNN equation for stereo vision, known to process a depth perception, to speech recognition. In real life speech recognition, it is considered that characteristic features of each phoneme are stored in our memory through daily life training and input speech data are compared with the memorized data to estimate their similarity (or disparity) for each phoneme. When the equation is applied to phoneme recognition, it develops competition among similarities of different phonemes and cooperation among neighboring frame data, and selects a specific phoneme as a recognized one, beating others down to zero. The TLNN was applied to the similarities with best 5 hypotheses among 24 kinds of phonemes. The average rates for speaker independent recognition were 78.05 % for 216 word database and 78.94 % for 240 word database by TLNN equation which were compared to 71.56 % and 72.37 % by HMMs, respectively.

## REFERENCES

1. S. Nakagawa "Ability and Limitation of Statistical Language Model" Proc.of ASJ:23-26,spring, 1998
2. Amari,S. and Arbib, M.A. "Competition and Cooperation in Neural Nets" Systems Neuroscience :119-165, Academic Press, 1977
3. D. Reinmann and H. Haken "Stereo Vision by Self-organization" Biol. Cybern. Vol.71:17-26, 1994
4. T.Kitazoe, J.Tomiyama, Y.Yoshitomi, and T.Shii "Sequential Stereoscopic Vision and Hysteresis" Proc. of Fifth Int.Conf. on Neural Information Processing, 391-396,October, 1998
5. ATR Japanese Speech Database and Technical Report, Japan, 1988
6. ASJ Continuous Speech Corpus for Research, Japan, 1991

## PITCH PERCEPTION IN THE JAPANESE MACAQUE: PHYSIOLOGICAL AND BEHAVIORAL APPROACH

Hiroshi Riquimaroux and Kazuchika Manabe

*hrikimar@mail.doshisha.ac.jp*

Department of Knowledge Engineering and Computer Sciences, Doshisha University,  
Kyotanabe, Kyoto 610-0321, Japan

### Abstract

We have investigated pitch extraction mechanisms in the central auditory system. Pitch perception is not always directly related to frequency or spectral information. A typical example is the missing fundamental. The missing fundamental perception is believed to be created in the central auditory system. Previous human studies have shown that the area around the primary auditory cortex appears to play an essential role in the missing fundamental perception. Previous animal behavioral researches have indicated that non-human animals also perceive the missing fundamental. Thus, neural correlates for the missing fundamental perception were studied in the primary auditory cortex of the Japanese monkey. We have found that the neurons in the primary auditory cortex, which is tuned to the fundamental frequency, also respond very well to a combination of successive higher harmonics of the fundamental frequency without the fundamental frequency itself. However, these neurons do not or little respond to each frequency component of the harmonics. The neurophysiological data basically agree well with the human psychoacoustical data. Our results suggest that the pitch perception originated from spectral information and the one created by time information have integrated at or below the primary auditory cortex. Next, we have stepped into experiments to investigate the dynamic pitch sequence which is related to music perception and speech communication. The monkey has been successfully trained to discriminate the direction of frequency and/or pitch change, rising or falling, by using tone bursts and complex tone bursts. How pitch information is processed in the central auditory system will be discussed.

**Key words:** auditory cortex, temporal pitch, spectral pitch, pitch sequence, monkey

### Introduction

Since 19th century, it has been a well known fact in the auditory psychology that we often perceive sounds which do not correspond to any of physical frequency components (spectrum) of the objective sound (Seebeck<sup>1</sup>, Schouten<sup>2</sup>). This is why we can enjoy music. For example, when we add several successive higher harmonics, we have a pitch sensation of the low-frequency fundamental which does not exist in the sound, called the "missing fundamental"(Fig. 1). It has been believed that the missing fundamental is created in the central auditory system above the superior olivary complex, where information from both ears meets for the first time. Previous data on human temporal lobectomy have indicated that the missing fundamental perception is mediated by the area around the primary auditory cortex (Zatorre<sup>3</sup>). Further, behavioral studies have shown that monkeys also possess the missing fundamental perception (Tomlinson and Schwarz<sup>4</sup>). So, we neurophysiologically and behaviorally studied the primary auditory cortex of the Japanese monkey (Riquimaroux et al.<sup>5</sup>).

### Physiological Experiments

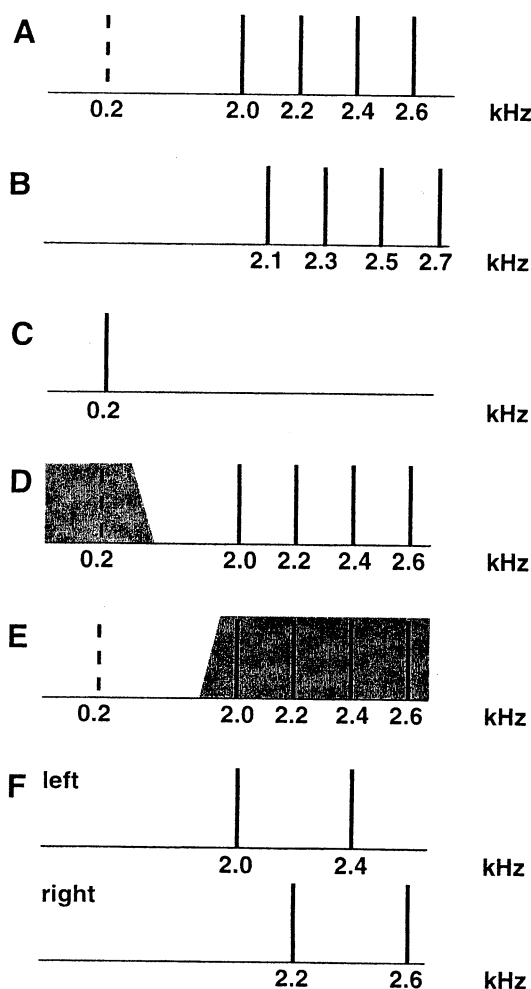
We examined whether information for the temporally created pitch (temporal pitch) known as the missing fundamental is handled by the same neuron in the primary auditory cortex that processes information for the spectrally created pitch (place pitch) based on the frequency components. Here, we used adult Japanese monkeys (*Macaca fuscata*), which have much more similar brain structure to humans than rodents or cats. Extracellular single unit recordings were made with glass-coated metal electrodes from the left primary auditory cortex on the superior temporal plane (Fig. 2). We have found that the neurons in the

primary auditory cortex, which is tuned to the fundamental frequency, also respond very well to a combination of successive higher harmonics of the fundamental frequency without the fundamental frequency itself. However, these neurons do not or little respond to each frequency component of the harmonics. The neurophysiological data basically agree well with the human psychoacoustical data (Fig. 3; Riquimaroux<sup>6</sup>). The findings suggest that the temporal pitch information appears to be processed together with the place pitch information by the same neuron in the primary auditory cortex. The data agree well with psychologically found evidences. Our

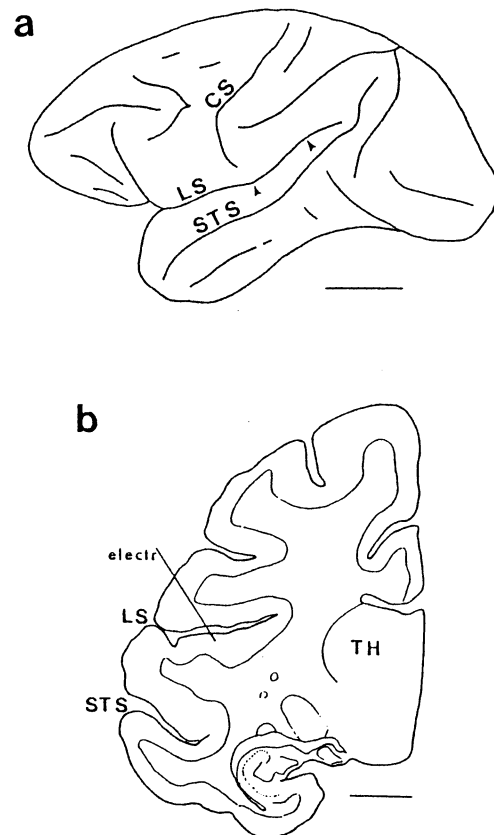
results may indicate that the pitch perception originated from spectral information and the one created by time information have integrated at or below the primary auditory cortex.

### Behavioral Experiments

Next, we have stepped into experiments to investigate the dynamic pitch sequence which is related to music perception and speech communication. A Japanese monkey was trained to discriminate the direction of frequency or pitch change, rising or falling, using water as reward (Fig.4; Manabe et al.<sup>7</sup>). The stimuli used were sequentially presented tone bursts and/or complex tone bursts. During the experimental sessions, the animal was kept in a monkey chair. When the animal put a hand through a hole to push a button, which was detected



**Fig. 1** Schematic power spectrum of synthesized waves. **A**, **B** and **C** correspond to a, b and c in Fig. 1. **D**: Low-pass noise is added to the wave in **A**. **E**: High-pass noise is added to the wave in **A**. **F**: Dichotic presentation of higher harmonics of 200 Hz. Odd harmonics to the right while even ones to the left. The missing fundamental is shown by a dotted line.



**Fig. 2** Recording sites in the cerebral cortex of Japanese macaque. **a**: Lateral view of the left cerebral cortex. The primary auditory cortex (AI) is located between two arrows within the lateral sulcus. The bar indicates 1 cm. **b**: A schematic electrode track shown in a frontal section of the left hemisphere. The bar illustrates 5 mm. **LS**: lateral sulcus. **STS**: superior temporal sulcus. **CS**: central sulcus. **TH**: thalamus. **electr**: electrode track.

by an optical sensor, the sound stimuli were presented. Once the animal take the hand away from the hole, the sound was not delivered. First, the animal learned to discriminate sequential frequency changes by using temporally paired tone bursts (Fig. 5). The intensity of each tone was randomly varied between 30 and 50 dB SPL to eliminate the possibility where the animal may use the intensity cue for the judgment. The animal could successfully discriminate the direction of tone burst sequence of various frequency pairs (Fig. 6). Then, the animal discriminated the direction of pitch sequence made of complex tones (Fig. 7) and that made of tone bursts and complex tone bursts (Fig. 8). The data may suggest that the monkey has a categorical perception for rising pitch and falling pitch similar to humans. We would like to examine which part of the cortical area plays a major role in controlling the sequential pitch perception.

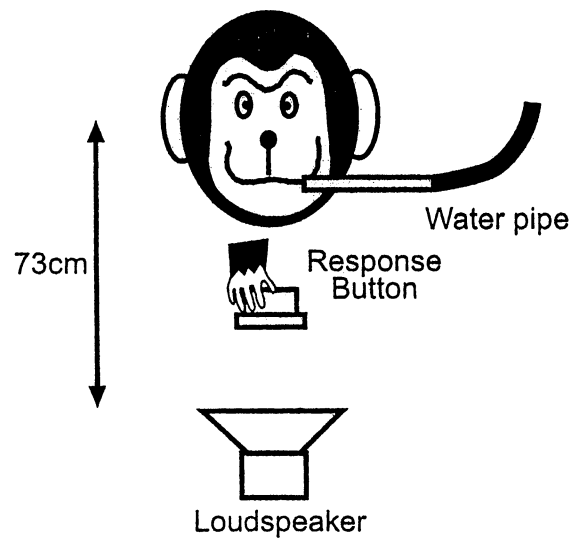


Fig. 4 Schematic experimental configuration.

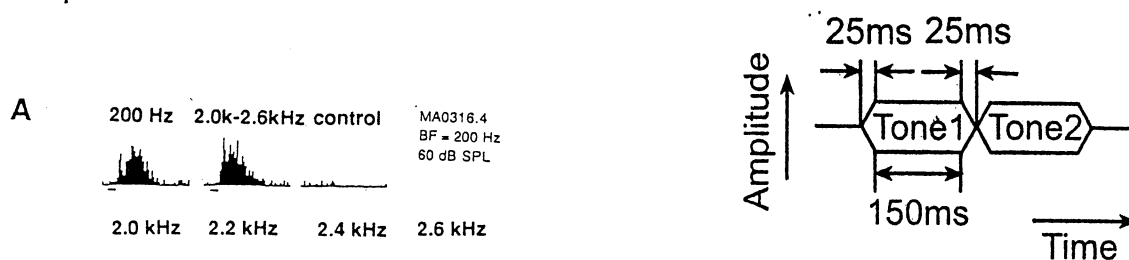
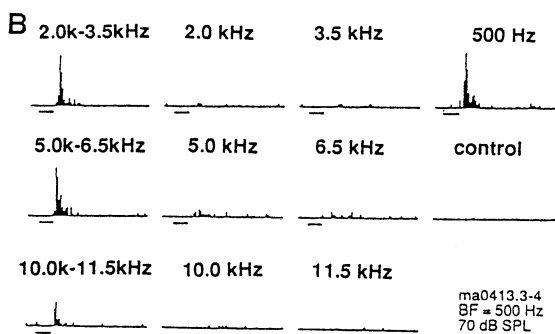
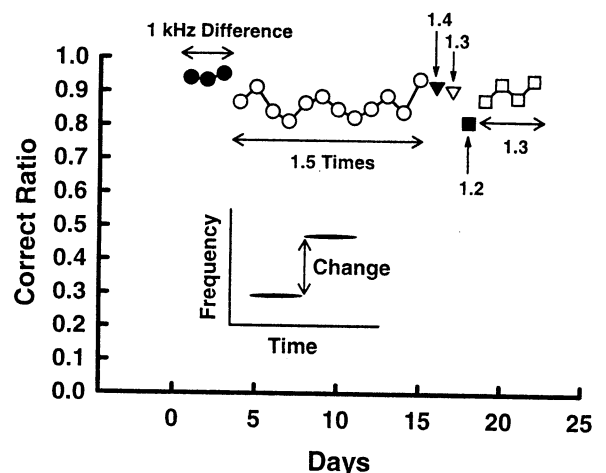


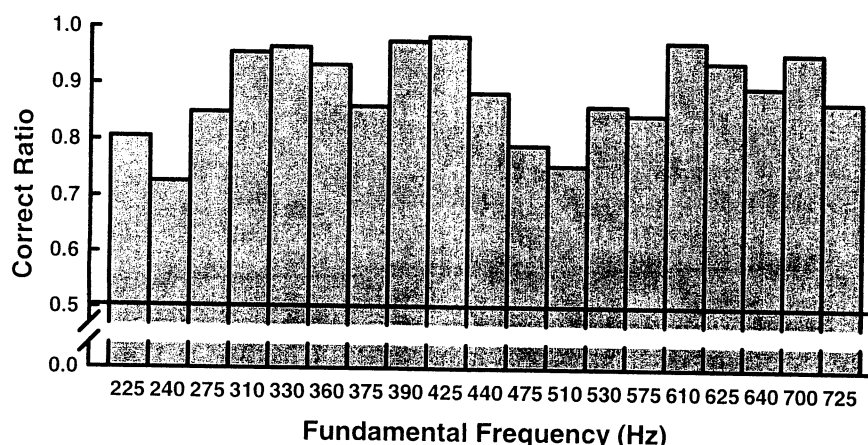
Fig. 5 Time pattern of sound presentation.



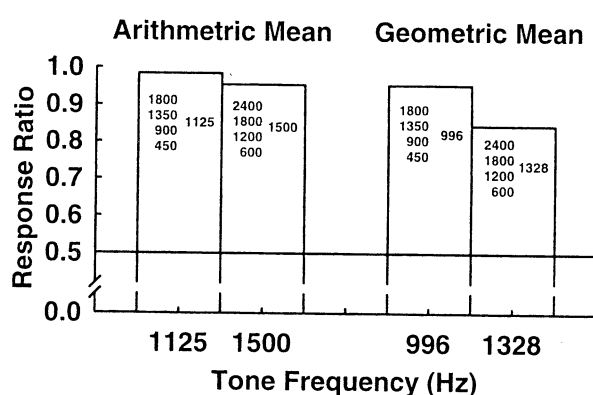
**Fig. 3** Responses to the fundamental frequency ( $f_0$ ), combinations of higher harmonics without  $f_0$  and each higher harmonic component. **A, top row:** from left responses to 200 Hz, a complex of 2.0 + 2.2 + 2.4 + 2.6 kHz and control. **bottom row:** responses to 2.0, 2.2, 2.4 and 2.6 kHz, respectively. BF = 200 Hz, Intensity = 60 dB SPL. **B, top row:** responses to a complex of 2.0 + 2.5 + 3.0 + 3.5 kHz, 2.0 kHz, 3.5 kHz and 500 Hz. **middle row:** responses to a complex of 5.0 + 5.5 + 6.0 + 6.5 kHz, 5.0 kHz, 6.5 kHz and control condition. **bottom row:** responses to a complex of 10.0 + 10.5 + 11.0 + 11.5 kHz, 10.0 kHz and 11.5 kHz. BF = 500 Hz. Intensity = 70 dB SPL.



**Fig. 6** Correct ratio for tone-tone discrimination. In the beginning, the difference in frequency was kept at 1 kHz. Then, the ratio between two tones were varied from 1.5 through 1.2.



**Fig. 7** Correct ratio for complex tone vs. complex tone discrimination. Fundamental frequency of lower complex tones is shown in abscissa. The fundamental frequency of higher complex tones was 1.5 times higher than the lower one.



**Fig. 8** Response ratio for complex tone vs. single tone discrimination. The frequency of the single tone was at the arithmetic average of frequencies of complex tone shown in inserts.

### Summary

Our neurophysiological studies using monkeys confirmed that the primary auditory cortex may play an essential role in the pitch information processing. Further, the present behavioral study implies that the monkey appears to have sequential pitch sensation just as humans do. In order to investigate the central pitch extraction mechanisms, behavioral studies with a focal reversible lesion (Riquimaroux<sup>8</sup>) and neurophysiological investigations with awake behaving animals are planned.

### Acknowledgement

This study was performed through Strategic Promotion System for Brain Science supported by Special Coordination Funds from the Science and Technology Agency of the Japanese Government.

### References

- 1) Seebeck, A. *Ann. Phys. Chem.* 53: 417-436 (1841).
- 2) Schouten, J. F. K. *ned. akad. Wet. Proc.* 41: 1086-1093 (1938).
- 3) Zatorre, R. J. *J. Acoust. Soc. Am.* 84: 566-572 (1988).
- 4) Tomlinson, R. W. W. and Schwarz, D. W. F. *J. Acoust. Soc. Am.* 84: 560-565 (1988).
- 5) Riquimaroux, H., Takahashi, T. and Hashikawa, T. *J. Acoust. Soc. Jpn. (E)* 15: 171-177 (1994).
- 6) Riquimaroux, H. *RIKEN Review* 9: 13-14 (1995).
- 7) Manabe, K., Toriyama, S. and Riquimaroux, H. *Proc. Acoust. Soc. Jpn. Fall Meeting*: 457-458 (1998).
- 8) Riquimaroux, H., Gaioni, S. J. and Suga, N. *Science* 251: 565-568 (1991).



# HANDS-FREE SPEECH RECOGNITION IN ECHO AND NOISE ENVIRONMENTS

*Sung-Il Kim . Tetsuro Kitazoe*

*Department of Computer Science and Systems Engineering*

*Faculty of Engineering , Miyazaki University*

*1-1 . Gakuen Kibanadai Nishi , Miyazaki . 889-2192 Japan*

## ABSTRACT

For some applications such as hands-free teleconferencing and telecommunication systems, the near-end speech signal to be transmitted is disturbed by ambient noise and by an echo due to the coupling between the microphone and the loudspeaker. Furthermore, the environmental noise including channel distortion or additive noise is assumed to affect the clean speech. In the present paper, we introduce a new approach using echo cancellation and Maximum A Posteriori (MAP) environmental adaptation technique to improve hands-free speech recognition accuracy. In this approach, it is shown that our proposed new system is effective for hands-free speech recognition. The experimental results also show that echo canceller and MAP adaptation are well adapted to the echo and noise environments.

## 1. INTRODUCTION

In the past few years, many works have been performed in HMMs to improve the speech recognition accuracy with a close-talking microphone. But especially recently, the dissemination of hands-free communication systems requires to provide users with some comfort. Therefore, problems of reverberant speech recognition have to be solved to obtain a good distant-talking speech recognition accuracy[1,2]. In hands-free speech recognition, since the microphone and the loudspeaker are coupled, the sound from the loudspeaker is picked up by the microphone, both directly and indirectly and this is heard by the recognizer as echo, causing unwanted speech recognition [3].

If the hands-free mode is to be used, we inevitably face with the problem of environmental noise including channel distortion or additive noise in addition to the acoustic echo which is fed back from the hands-free mode. Furthermore, the clean speech is affected by channel noise. The observed signal is

$$y(t) = s(t) * h(t) + n(t) \quad (1)$$

where  $n(t)$  is the total additive noise and  $h(t)$  is the channel mismatch. Figure 1 shows how the convolutional noise and the additive noise are assumed to affect the clean speech. In the conferencing application as shown in figure 2, hands-free microphone pick up at the remote end includes both the desired speech and the undesired ambient

room noise such as air conditioning, heating, or ventilating systems and computer fans, etc.

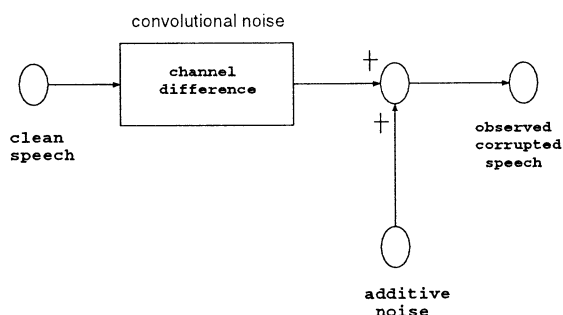


Figure 1: Channel distortion and additive noise environments

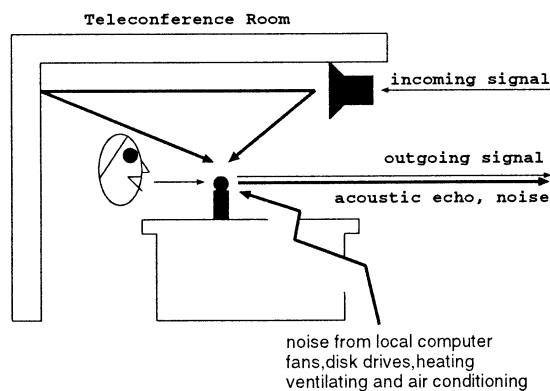


Figure 2: Echo and ambient noise environments

Furthermore, when the speech, for instance, will be mixed with any noise present within the vehicle, echo cancellers generally find it difficult to eliminate echo if there is a high level of background noise. In our experiments, we extend the MAP estimation[4] to an environmental adaptation[5] for the recognition of hands-free speech signal obtained from the laboratory environments which have channel distortion or additive noise.

In automobile application area, for instance, the hands-free microphone picks up only the desired speech and removes the undesired echo by using acoustic echo canceller and then the speech recognizer improves the echo-cancelled speech recognition accuracy by using environmental adaptation technique in a channel distortion and additive noise environment of a car.

In this paper, we report an implementation of new approach for hands-free speech recognition using echo canceller and MAP adaptation technique. We show that the proposed system is effective for hands-free speech recognition. We also report hands-free speech recognition rates in comparison with close-talking ones.

## 2. ECHO CANCELLER AND ENVIRONMENTAL ADAPTATION

Acoustic echo was first encountered with the early video/audioconferencing studios and also occurs in a typical mobile situation, such as in a car. In this situation, sound from the loudspeaker is heard by the listener, as intended. However, this same sound is also picked up by the microphone, both directly and indirectly. The result of this reflection is the creation of multipath echo and multiple harmonics of echo encountered, which unless there are eliminated, are transmitted back to the distant end and is heard by the talker as echo.

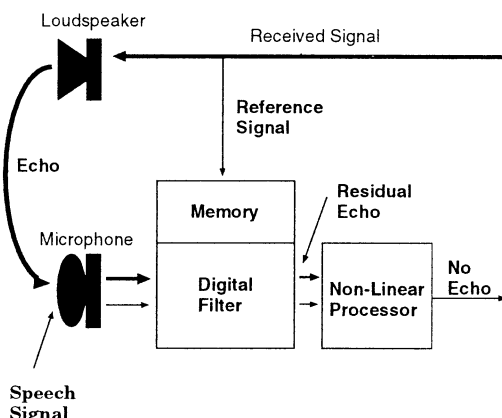


Figure 3: Operation of an acoustic echo canceller

Figure 3 shows the basic operation of an acoustic echo canceller in a conference room type of situation. In echo cancellation, complex algorithmic procedures are used to compute speech models. This involves the system generating the sum from reflected echoes of the original speech, then subtracting this from any microphone signal it picks up. The result is the purified speech of the person talking. The format of this "echo prediction" is learned by the echo canceller in a process known as adaptation. It might be said that the parameters learned from the adaptation process generate the prediction of the echo signal, which

then forms an 'audio picture' of the room in which the microphone is located.

During the conversation period, this audio picture constantly alters, and, in turn, the canceller has to adapt continually. The time required for the echo canceller to fully learn the acoustic picture of the room is called the convergence time. Other important performance criteria involve the acoustic echo canceller's ability to handle acoustic tail circuit delay. This is the time span of the acoustic picture and roughly represents the delay in time for the last significant echo to arrive at the microphone. Another important factor is acoustic echo return loss enhancement(AERLE). This is the amount of attenuation which is applied to the echo signal in the process of echo cancellation i.e., if no attenuation is applied, full echo will be heard. The canceller's performance also relies heavily on the efficiency of a device called the center clipper, or non-linear processor. This needs to be adaptive and has a direct bearing on the level of AERLE that can be achieved.

The MAP estimation is also called bayesian successive estimation of the HMM parameters for the new speaker in a framework. The estimated mean vector value after given N samples is shown as

$$\hat{\mu}_N = \frac{\alpha\mu_o + \sum_{i=1}^N X_i}{\alpha + N} \quad (2)$$

where  $\alpha$  is an adaptation parameter. The estimated covariance matrix by using N samples is

$$\begin{aligned} \hat{\Sigma}_N &= \frac{1}{\beta + N} \{X_N X_N^T - (\alpha + N)\mu_N \mu_N^T \\ &+ (\beta + N - 1)\Sigma_{N-1} \\ &+ (\alpha + N - 1)\mu_{N-1} \mu_{N-1}^T\} \end{aligned} \quad (3)$$

where  $\beta$  is a coefficient. In our experiments, the values of  $\alpha$ ,  $\beta$  were set at 15 and 50 respectively, which were determined experimentally.

Speaker adaptation is performed with successive training of SI models using small amounts of adaptation speech data. There are two adaptation methods. One is well-known as the supervised speaker adaptation(SSA) which achieves the adaptation in accordance with correct label sequences. Another method is known as the unsupervised speaker adaptation(USA). In this method, the label sequences are provided automatically by the viterbi segmentation as shown in figure 4. Therefore, a block diagram of the overall hands-free speech recognition system based on acoustic echo canceller and MAP environmental adaptation is shown in figure 5.

## 3. EXPERIMENTAL CONDITIONS

We trained Japanese 40 phoneme HMM models using 5240 labeled word utterances of 10 male speakers and 503 sentences of 6 male speakers in the ATR Japanese speech database. In the test, we used ATR 100 phoneme balanced

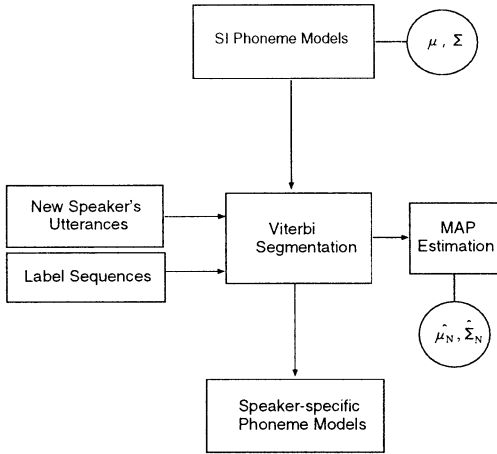


Figure 4: Block diagram of unsupervised speaker adaptation

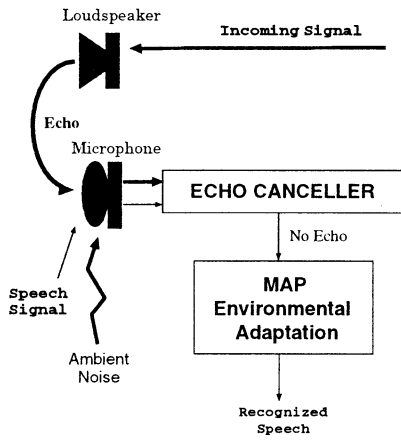


Figure 5: Overall block diagram of hands-free speech recognition system

words. A set of 20 dimensional observation sequences including discrete duration information are obtained for recognition. Table 1 shows the preprocessing analysis condition of the speech data and model topology.

Figure 6 shows the experimental conditions for each speech-talking modes. In this figure, the clean mode is the method using the models trained on clean speech. In the close-talking and hands-free mode, speech data were also recorded using clean speech in a laboratory room environments, including ambient noise such as air conditioning systems, fans and disk drives, etc. In hands-free mode experiments, we used a station noise data from the noise database of Japan Electronic Industry Development Association (JEIDA) as echo controlled by digital audio tape recorder. The speech signals in hands-free mode were captured by using a desk-top microphone which was located at a distance of 1.0-1.5m from the loudspeaker in the echo

Table 1: Analysis of speech signal

sampling rate	16kHz.16bit
preemphasis	0.97
window function	16 msec Hamming window
frame period	5 ms
feature parameters	10-order MFCC +10-order delta MFCC +log power+delta log power +discrete duration information
model topology	3 state left-light phone model

and noise environments.

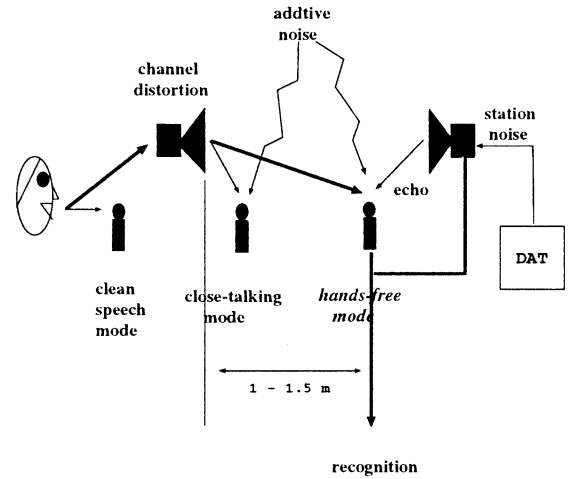


Figure 6: Experimental conditions for each speech-talking modes

## 4. EXPERIMENTS AND RESULTS

Table 2 illustrates the recognition rates of clean and close-talking modes, in terms of the number of adaptation words. As can be seen, the clean mode speech recognition rates outperformed the close-talking mode ones. The experimental results also show that the clean mode had just 10% improvement in recognition rates in accordance with the increase of the number of adaptation words, whereas in the close-talking mode, it showed 76% dramatic improvement in recognition ones by increasing the number of adaptation words.

Table 3 illustrates the recognition rates of hands-free mode when using echo canceller and not using one. When the number of adaptation words was increased to 50, we were able to improve the recognition rates by more than 20% by using echo canceller in comparison with non-echo one in hands-free mode.

Table 2: Recognition rates of clean and close-talking modes

Number of Adaptation Words	Mode	
	Clean	Close-talking
baseline	85	5
7	87	35
13	90	45
25	94	71
50	95	81

Table 3: Recognition rates of hands-free modes

Number of Adaptation Words	Hands-free mode	
	+echo canceller	non-echo canceller
baseline	2	1
7	8	14
13	20	14
25	29	18
50	41	19

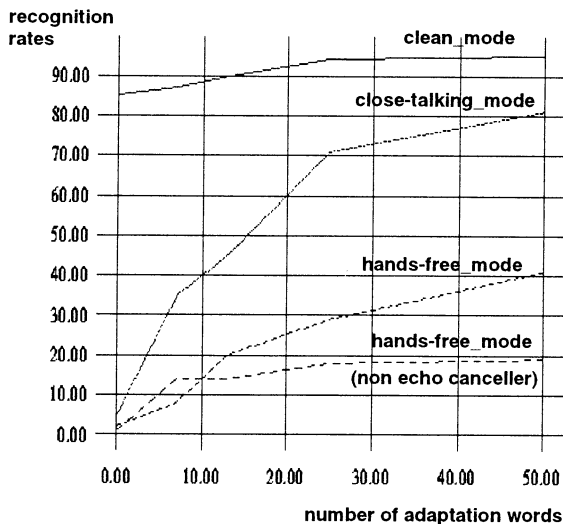


Figure 7: Recognition rates of three kinds of speech signal modes

Figure 7 summarizes the performance comparison of several speech signal modes with a plot of word recognition rates versus number of adaptation words. Though the recognition rates of the hands-free mode are lower than those of the other modes, owing to the over-cancelled speech signal and high level echo and additive noise, it shows us that the combination of echo canceller and MAP adaptation technique is effective for hands-free speech recognition. Particularly, the experimental results also show that each modes are rapidly adapted to the model parameters of echo and noise environments by MAP estimation.

## 5. CONCLUSIONS

This paper described an efficient method of hands-free speech recognition based on the use of acoustic echo canceller and MAP environmental adaptation technique. The experimental results indicated that the hands-free mode using echo canceller and MAP environmental adaptation showed the possibility of the recognition rate improvement in the echo and noise environments. The results also showed that the speech models using echo canceller and MAP adaptation were well adapted to the echo and noise environments. In the future, we will concentrate on the recognition rate improvements for hands-free speech recognition in task-specific applications.

## References

- [1] D. Giuliani, M. Matassoni, M. Omologo and P. Svaizer : "Hands Free Continuous Speech Recognition In Noisy Environment Using a Four Microphone Array", Proc. ICASSP, pp. 860-863, 1995
- [2] T. Takiguchi, S. Nakamura, Q. Huo and K. Shikano : "Model Adaptation Based on HMM Decomposition for Reverberant Speech Recognition", Proc. ICASSP, pp. 827-830, 1997
- [3] R. Martin and J. Alenhoner : "Coupled Adaptive Filters for Acoustic Echo Control and Noise Reduction", Proc. ICASSP, pp. 3043-3046, 1995
- [4] Y. Tsurumi and S. Nakagawa : "An Unsupervised Speaker Adaptation Method for Continuous Parameter HMM by Maximum A Posteriori Probability Estimation", Proc. ICSLP, pp. 431-434, 1994
- [5] J.H. Lee, B.K. Kim and H.Y. Chung : "Environmental Adaptation Using A Posteriori Estimation for Korean Word Recognition", Proc. IEEE Invited Workshop on Pattern Recognition for Multimedia Techniques, pp. 49-52, October, 1996

## Artificial Mind in a Classical Context

Scott HAGAN and Masayuki HIRAFUJI

*Computational Modeling Lab, Dept. of Information Science, N.A.R.C.  
3-1-1 Kannondai, Tsukuba, Ibaraki, Japan 305-8666*

### Abstract

Can artificial minds emerge from complex information processing systems and under what dynamical assumptions? We explore fundamental physical restrictions that limit the formulation of artificial minds through thought experimental arguments in classical physics. Emergence must incorporate dynamical criteria to answer questions about when, how and what becomes the property of consciousness. The resulting accounts rely on apprehending global states and classically must be reconciled with two kinds of relativity. Relativity with respect to the observer's frame can be accommodated by requiring that the emergent property be Lorentz invariant. Relativity with respect to the system boundary can only be reconciled with the ontological character of mental states if these states are intrinsically defined. Local, causal dynamics must be continuously updated as to the location of an intrinsic boundary, a requirement that violates locality constraints. Since there is no pre-existing, ordered space in which mental content can incrementally accrue, consciousness cannot use locality relations to compose perceptual space. Local representations in physical space cannot be made sufficiently complex and distributed representations cannot make the encoded information simultaneously available without contravening locality. The conclusion, that classical science cannot account for all features of conscious processing, finds convergence in Penrose's recent thesis and might likewise be circumvented in quantum theory.

Keywords: artificial mind, emergence, locality

### Introduction

Characteristic features of human mentation suggest that enormous processing advantages might be realized if strategies employed by conscious agents could be artificially reproduced. We explore here some fundamental physical limitations that apply to theories of artificial mind by examining some thought experiments in classical physics.

Theories that directly identify mental features with properties that are physically reducible are guilty of a category error: mental states cannot be identical to physical states unless the two are indiscernible. Theories that scrupulously preserve the distinction between mental and physical, on the other hand, run the risk of leaving mental constructs incapable of causally influencing physical processes. Emergence has arisen, particularly in complex systems theory, as the prime candidate to circumvent both kinds of error. The purpose of this note will be to establish that no emergent account of conscious mental states is viable within the limits afforded by classical science.

### Emergence

The allure of emergence is its ability to accommodate discourse about mental properties without incurring category error. Conflation of the mental and physical ontologies is avoided by postulating a hierarchy of levels, in which "higher-order" properties, corresponding to mental features, arise at some threshold level of complexity.<sup>1</sup> To then answer questions about what exactly should become

the property of consciousness, when the emergent mechanism should come into play and why it should be seen as a necessary consequence of the concomitant conditions, one must distinguish which varieties of emergent explanation are compatible with a classical scientific characterization.

Any explanation of mental states in the framework of classical science must accommodate the *supervenience* of emergent properties on properties at the level of microphysical dynamics. Emergent properties supervene on microphysical properties whenever two entities, identical at the microlevel, are of necessity also identical with respect to all emergent properties. Supervenience does not impose the additional, unnecessary constraint that each emergent property should be uniquely associated to a single microphysical configuration. The classical picture further requires that the dynamics at the microphysical level be deterministic. The nature of explanation in the classical model entails that there exist some criteria, possibly inaccessible to empirical observation, on the basis of which the system evolves. This does not in practice imply predictability, nor does it imply that higher-order descriptions must be reducible to expressions in terms of lower-level processes. A definition of emergence along these lines is, for instance, engaged in discussions of chaotic phenomena in complex systems.

With this refined concept of emergence, it is possible to specify dynamical criteria on the basis of which answers to questions about what, when and how might be ventured. Models of sensory perception have

already speculated about the nature of these criteria, variously identifying perceptual contents with “vector activations”, “concentric epicenters” or “attractors.”<sup>2</sup> What these accounts have in common is that they all impute mental properties to global states. This is the only option available to an emergent classical theory since any property that can be identified with local states is completely reducible to physical entities.

Emergent properties so constituted are locally implicit, as guaranteed by supervenience, but are not required to be explicitly represented at the system level, available and empowered to influence causal dynamics locally. For this reason emergent features are generally characterized as extrinsic. Rendering them explicit requires an act of an observer, external to the system itself, apprehending the whole.

### Relativity in Extrinsic Accounts

If mental states are extrinsic, then they must be reconciled with the relative nature of an account given in terms of external observers. The singular character of real conscious experience must be compatible with accounts given from all possible external vantages.

How this might be accomplished can be illustrated in terms of the standard relativity with respect to frame, by a *gedanken* experiment in special relativity. Observer ‘A’ in the rest frame of a conscious entity will characterize a mental state of that entity as some emergent property of a spatially extended system. Each state is given as a spatially bounded set of events with the same temporal coordinate and from these the observer can create a time series of mental states, the ‘thoughts’ of the conscious entity according to ‘A’. Another observer ‘B’, in a frame moving with fixed relativistic velocity with respect to the conscious entity, likewise formulates a time series of mental states but, since the definition of simultaneity differs in this frame, will compose each mental state from a different collection of events. The ‘thoughts’ of the conscious entity according to ‘B’ need not, *a priori*, match up with those recorded by ‘A’. It is clear that, should the accounts differ, only one (at best) can be correct since only one mental series actually occurs consciously.

There are two possible resolutions in this case. First, one could require that the time series composed by each possible observer be equivalent to every other. This amounts to imposing a non-trivial constraint, Lorentz invariance, on the formulation of the emergent property as instantiated in terms of physical states. This approach is consistent with both an extrinsic definition of conscious mental states and with the singular quality of actual mental sequences.

Second, one might rule that all accounts from observers outside the rest frame are incorrect. According to this option, the only correct account that

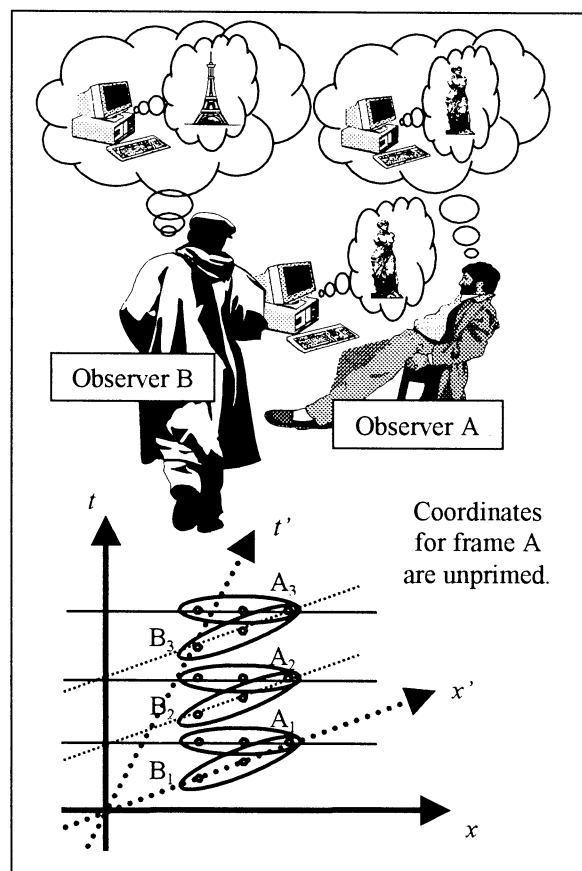


Figure 1: On an extrinsic account, observer ‘A’ in the rest frame of a ‘conscious’ computer extracts a series of mental states from the set of simultaneous events,  $A_1, A_2, A_3, \dots$ . For a relativistic observer ‘B’ (depicted as a ‘fast’ walker), the series of mental states corresponding to the same physical system is extracted from a different set of simultaneous events,  $B_1, B_2, B_3, \dots$  and, if no invariance is imposed, the sequence will in general differ. Both accounts cannot be correct.

need be given is the one formulated in the rest frame. The determination of content in conscious states is demoted to the level of a perspectival artifact. This might plausibly be maintained only if an extrinsic definition of the emergent state is abandoned. The conscious entity does not, on the view adopted, access and implement one or another of the possible extrinsic accounts. If this were the case, then the account in the rest frame might indeed claim priority over all others. But since the consciousness of the entity has been *defined* extrinsically, there should be no issue of access and no priority of particular cases. What is required then is a shift to an intrinsic definition in which the emergent property is explicitly represented at the system level. Since the conscious system is always trivially in its own rest frame, this gives that frame the desired priority.

To resume, the potential multiplicity of accounts can be accommodated by either (i) imposing Lorentz invariance on an extrinsic formulation or (ii) adopting the perspective of the rest frame in an intrinsic

formulation. Relativity is thus resolved in extrinsic theories and repudiated in intrinsic theories.

A second form of relativity to which an emergent characterization is subject is a relativity with respect to the boundary defining the separation of 'system' and 'environment'. This is, in the context of classical science, a boundary chosen arbitrarily to reflect which system in particular it is desirable to characterize. Since the emergent properties under discussion recognize information and/or relationships available only at the global level, their characterization will be sensitive to a specification of the boundary through the operative definition of "global." Though certain systems will inevitably be described more conveniently, classical phenomena do not in general insist on being circumscribed by particular boundaries. The use of the term 'system' in classical science reflects this arbitrariness and the absence of a preferred characterization.

Conscious mental states, in contrast, are not arbitrary. They encompass a certain scope, particular content. Two researchers studying the same conscious entity with different definitions of what exactly constitutes the conscious system might, *a priori*, determine different conscious states. One can, for instance, imagine that the first researcher defines the system boundary to enclose the smallest physical area (or one of the smallest) in which mental properties are manifest while a second researcher chooses a boundary so as to include the first system as a subset. The accounts of conscious mental states given by these researchers can presumably be arranged to differ in content but the actual mental state of the studied entity cannot simultaneously satisfy two differing ascriptions.

For comparison we contrast this case with that of another property, 'liquidity', often cited as a paradigm case of emergence. Again we find that systems characterized as 'liquid' are not uniquely defined. The contents of a full glass of water manifest the property, but so too does the bottom half of the water. Here though there is no contradiction in describing 'liquidity' as an emergent property in both systems.

One might try to salvage an extrinsic account by imposing an invariance, analogous to Lorentz invariance in the case of relativity with respect to frame, that would ensure that all choices for the boundary lead to the same mental state. That this is impossible is most trivially evident if we imagine one researcher delineating the system boundary around the 'brain' of a conscious entity (here the 'brain' need not be more critically characterized) while a second observer draws her own boundary circumscribing the 'brains' of *two* conscious entities.<sup>3</sup> Invariance then implies that the consciousness of the first entity should be equivalent to the combined consciousnesses of both entities. This, obviously, is not the case.

## Locality in Intrinsic Accounts

If an emergent formulation is indeed appropriate, consciousness is distinguished from other emergent phenomena in being independent of whether and how an external observer extracts from the relevant facts about the system and a specification of its boundary, an interpretation that merely renders conscious states. Since no extrinsic account is viable, this independence is possible only if the system intrinsically recognizes that the conditions of satisfaction for a conscious state to emerge are met only with regard to a particular boundary.

Emergent properties are necessarily defined over global states and in order for them to have causal consequences, the global state must inform the local dynamics where the causal mechanisms in classical science are located. Even if the microphysical level is not seen as the sole repository for causal dynamics, an explanation of conscious states compatible with supervenience must allow that there is an interpretation of all causal processes at this level. If the emergent property is to be promoted to intrinsic status, the global state must inform the local dynamics of the current location of its boundary, along with any other information pertinent to its specification or the determination of causal consequences.

In a classical context, locality forbids the fulfillment of this requirement. Locality restricts the speed of propagation of signals over distances, however small. An explicit representation of the boundary cannot be made available to all points in an extended physical system simultaneously. This is why intrinsically defined systems are nowhere else invoked in classical science.

One might imagine dodging this objection by loosening the requirement of simultaneity and allowing information about the global state and its boundary to locally accrue. This strategy requires that physical localities individually assume the burden of storing an explicit representation of all causally empowered features of the global state including a specification of its boundary. This vastly overestimates the number of degrees of freedom available at individual localities in classical theory, especially once thermal noise is accounted for. Even conservative estimates of the capacity of simple, primitive states far outstrip the potential.<sup>4</sup> There is, moreover, reason to suspect that even these estimates underestimate the requirements.

Take visual perception as an example. The phenomenal space in which we experience the world visually is ontologically real, not a mere theoretical device, so it is not acceptable in a fundamental theory to assume a pre-existing structure. Some account of this structure must be given, just as cosmology must account for the structure of physical space-time.

Structure in the visual perceptual field is evident in ordering relations like “beside” or “above.” Despite the ‘spatial’ labels, these are not relations in physical space so the mental relations are not explained simply by the fact that physical entities are correspondingly related.

Part of the task of explaining conscious states is thus the construction of structured mental spaces. This however cannot be accomplished by accrual of information. Assembling mental structure piecemeal assumes a pre-existing space in which assembly takes place and which preserves the relations of assembled pieces (regardless of what the ‘pieces’ are thought to consist in). No locality relations exist in visual perceptual space until the space itself has been constructed so they cannot be used to facilitate the construction. A similarly vicious circle is encountered in cosmology: physical space-time cannot be assembled bit-by-bit using relations between points. There is no existing concept of locality by which to make sense of these relations. A sensible theory of space-time structure is necessarily non-local. A local account is nonsensical; it uses in the explanation the relations that are to be explained.

Mental spaces must similarly be constructed in a non-local way (with respect to mental space). All the necessary relations of mental primitives must be given explicitly and at once. Construction of structure cannot proceed on the basis of partial information. To make use of partial information, there must already be mental structure to store relations as data accumulate. All the data must therefore be simultaneously available in physical space (if mental structure is to derive from relations in physical space).

The classical paradigm provides two options for encoding this information in the physical substrate: local or distributed. A local encoding of all the relations between each mental element and every other cannot be accommodated even under the most absurdly minimal assumptions about the complexity of mental states. And for a distributed representation, locality once again forbids simultaneous access to information that is dispersed over an extended physical region, however small.

### Conclusion

We conclude that no emergent account of mental states appears to be possible within the context of classical science. If the emergent property is formulated extrinsically, it is impossible to reconcile the relativity in the definition of the system boundary with the singular nature of experienced conscious states. If it is formulated intrinsically, the locality constraint forbids a spatially extended global system from simultaneously accessing sufficient information to realize a mental state that is intrinsically bounded, complex and ordered.

Evidently the arguments raised with respect to intrinsic formulations are not universal. Intrinsic theories of modest complexity might be salvaged if classical systems with a large number of local degrees of freedom could be sufficiently shielded from thermal noise to allow vast numbers of states to be distinguished. There seems little evidence however that the strategies employed biologically make use of entirely local representations. The locality constraint preventing instantiation of physically extended intrinsic representations applies only within the context of classical theory and might be circumvented in theories admitting non-classical elements in a non-trivial way. Penrose’s non-computational processes<sup>5</sup> may qualify as would quantum theories if macroscopic quantum states prove, as now appears likely, to be intrinsically bounded. It is in practice problematic to realize such states, and difficult to see what role they could play biologically<sup>6</sup>, but if the arguments made here are valid, then such possibilities may hold an essential piece of the puzzle.<sup>7</sup>

<sup>1</sup> See for instance Sperry RW (1992) Turnabout on consciousness: a mentalist view. *Journal of Mind and Behavior* 13:259-280.

<sup>2</sup> Respectively Churchland PM (1995) *The Engine of Reason, the Seat of the Soul: A Philosophical Journey into the Brain*. MIT Press: Cambridge, MA; Greenfield SA (1995) *Journey to the Centers of the Mind*. W.H. Freeman: New York; Port RF, Van Gelder T (eds.) (1995) *Mind As Motion: Explorations in the Dynamics of Cognition*. MIT Press: Cambridge, MA.

<sup>3</sup> Rosenberg GH (1997) *A Place for Consciousness: Probing the deep structure of the natural world*. Ph.D. dissertation, Indiana University.

<sup>4</sup> Marshall IN (1989) Consciousness and Bose-Einstein condensates. *New Ideas in Psychol.* 7(1):73-83.

<sup>5</sup> Penrose R (1994) *Shadows of the Mind*. Oxford University Press: Oxford.

<sup>6</sup> Some possibilities are considered in Jibu M, Pribram KH, Yasue K (1996) From conscious experience to memory storage and retrieval. *Int. J. Mod. Phys.* B10:1735-1754; Hameroff SR, Penrose R (1996) Conscious events as orchestrated space-time selections. *J. Consciousness Studies* 3:36-53; Jibu M, Hagan S, Pribram K, Hameroff SR, Yasue K (1994) Quantum optical coherence in cytoskeletal microtubules. *BioSystems* 32:195-209.

<sup>7</sup> See also Stapp HP (1995) Why classical mechanics cannot naturally accommodate consciousness but quantum mechanics can. *Psyche* 2(5).



## A Computational Model of Viewpoint-Forming Process in Searching Solutions - Theory and Methods using Hierarchical Classifier System -

Takahiro Yoshimi

RACE: Research into Artifacts.  
Center for Engineering  
The Univ. of Tokyo  
4-6-1 Komaba, Meguro-ku.  
Tokyo 153-8904, Japan

Toshiharu Taura

RACE: Research into Artifacts.  
Center for Engineering  
The Univ. of Tokyo  
4-6-1 Komaba, Meguro-ku.  
Tokyo 153-8904, Japan

### Abstract

In an adaptive machine learning system, it is inevitable that the production rule (*if-then* rule) has a long *if*-part when input information is highly complex and requires an abundance of properties. In such an adaptive system, without using some method to look only at specific positions and ignore others in the rule, adaption to external dynamical environment changes usually results in a very inefficient process. In order to avoid such a result and to systematize the method in terms of its knowledge structure, which is expressed as a rule, the authors present a computational model of what the authors call the Viewpoint-Forming Process. The key concept of this proposal, *Viewpoint*, contributes to focusing on adequate properties within input the information, by means of using an expressed knowledge, instead of the automatic focusing embedded in the adaptation process. A hierarchical classifier system is applied to facilitate the realization of the computational model called the Viewpoint-Forming Mechanism. A computer system is developed to evaluate the utility of the Viewpoint-Forming Mechanism, using a simple and easily understood path-finding problem as an example.

Key Words: viewpoint, classifier system, genetic algorithm, production rule, don't care

### 1 Introduction

The use of machine learning system has been an effective method of acquiring knowledge. However, the fields where such machine learning systems are required are becoming increasingly more and more complex. Under such a complex environment, in order to acquire effective knowledge in terms of its generality and re-usability, (1) a system that can deal with more possibilities of external environment changes must be designed, but (2) knowledge required for such a purpose is inclined to be more complex in general.

A machine learning system which acquires knowledge by using the production rule (*if-then* rule) is one of the most general types of systems. For this kind of system, in this research, the authors solve (1) above by using Classifier System (CS) proposed by Holland[1]. The CS has a mechanism which can perform genetic operation on a production rule (*classifier* as shown in Figure 1), and can acquire the knowledge desired. Also the CS can adapt to dynamic changes of the environment comparatively easily by performing genetic operations(see, for practical example Wilson[2], Hilliard et al.[3], and Goldberg[4]). Thus, for CS, complete control that forecasts all behaviors of the system is not necessary.

```
<classifier> ::= <condition>:<action>
if <condition> then <action>
<condition>={0,1,#}Lcondition
<action>={0,1}Laction
```

Figure 1: Definition of *classifier*

On the other hand, as mentioned in (2), the *if*-part of rules is inclined to be more complicated when the environment of the system is complex. If such complicated information can be systematized for each case, a hierarchical rule can be designed. However, it is rare to systematize information into order at the stage of designing rules. Therefore it is common to use the rule in which each independent property is simply allocated sequentially in both input information and the *if*-part of rules. Although this is comparatively easy, the more properties are considered in input information, the longer the *if*-part of the rules, as shown in Figure 2. Added to this, since the *then*-part follows the *if*-part, the more complex the environment becomes, the longer the length of rules.

Although the use of the CS can answer the (1), it is disadvantageous due to the extension of the length of rules caused by (2). That is, in an environment which

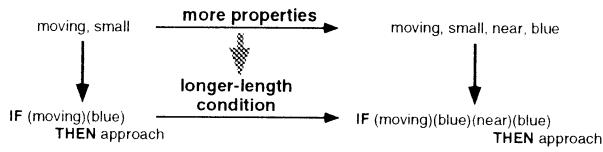


Figure 2: more properties and the longer length of rule

requires long rules, the structure of the phenotype become complex. Consequently the solution space of the genotype will expand as shown in Figure 3, making it more difficult to find a solution effectively by using genetic operations which cause emergent behavior.

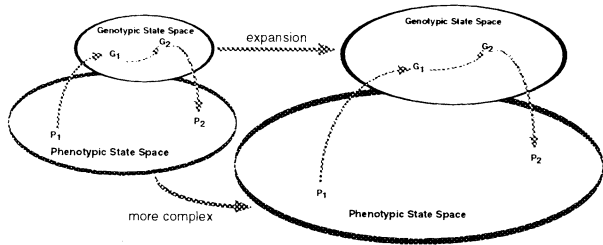


Figure 3: Phenotypic/Genotypic state space in GA

On the other hand, not all properties considered at the stage of designing rules are always important in such complex environments. Sometimes only a few properties need be looked at, and such limitation contributes to a faster determination of a solution. In the original CS developed by Holland, in order to avoid the ineffectiveness in finding a solution, the symbol # is ordinary used. This symbol is usually called "Don't care" and can be substituted by either 0 or 1. Holland mentioned that # is "useful for looking at some specific positions and ignoring others" [5]. This implies that using # and ignoring the information at that position is, in a sense, a kind of abstraction. However, the position of # is fixed as a result of hundreds of genetic operations. In other words, the knowledge of how to fix the position of # is embedded in the evolutionary process and iteration of operations, of input messages: thus it will not appear until the solution is found. From this, it follows that, in a classical CS, finding a solution and its abstraction are performed in parallel.

Because it would contribute to the clarification of the mechanism of abstraction, the authors believe that the way of fixing the position of # is highly important in cognitive systems based on the computational model, and the knowledge for fixing the position should be better described for systematization of the process. To observe the process of abstraction apart from finding the solution itself, another computational model for the process must be assumed.

Therefore in this paper, the authors propose the computational model of the Viewpoint-Forming process that uses classifier system. In the system capable of the Viewpoint-Forming process, the authors aim at expressed description of knowledge (which is the same as "rule" within the system) to look at specific positions and ignore others in the rule. Hereafter, the authors calls the rule for finding a solution which is ordinarily used in the classical CS, the Solution-Finding Rule (SFR), to distinguish it from the rule for forming a viewpoint (Viewpoint-Forming Rule: VFR).

The rest of this paper is organized as follows. First, in section 2, the authors give an explanation of the concept of *Viewpoint*, and also the basic idea of CS in order to explain the concept of Hierarchical Classifier System (HCS). Section 3 follows with a description of a method for the concrete implementation, as well as an example problem considered by us and the results. Finally, the authors summarize and draw conclusions based on the results of our study, in section 4.

## 2 Realization of Viewpoint-Forming Process

### 2.1 Concept of Viewpoint

In a production system which consists of *if-then* rules (In this paper, it is called as "SFR") with adaptive processing, the length of the *if*-part increases as it has to consider more properties, which sometimes hinders the system's adaptation speed. In particular, in CS, irrespective of whichever smart encoding method the system designer may create, it cannot be shortened more than basic limitation. For a simple example, a 3 bits condition can not express more than 8 different values.

In order to solve the problem of decreased system effectiveness when put into practical use due to the increasing length of the *if*-part as more properties are considered, the authors propose the method which uses only certain property of the rule as required according to the scene, as Viewpoint-Forming Mechanism (VFM). That is a method which forms the viewpoint in order to select the property required in that particular scene. Due to the fact that the input information and *if*-part of SFR may consist of several properties, the system looks at only certain property part required as determined by the Viewpoint-Forming Rules (VFR). Figure 4 shows the process concept of the Viewpoint-Forming in production rule and a simple example of the processing. In the figure, 6 properties exists in the SFR's *if*-part, and when the system faces the scene A, it looks at and compares only the

second and the fourth properties, and while facing the scene B, only the first and third. On facing some special scene C, the system looks at and compares all the properties. As an alternative to comparing all the properties correctly only by means of the SFR, the authors expect a quicker adaptation by the VFM which is adequate for the scene.

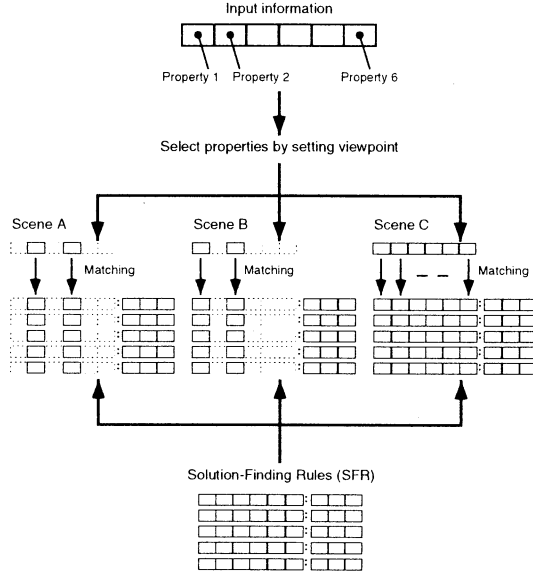


Figure 4: Viewpoint-Forming in production rule

However it appears to be difficult to set the adequate property of the rule. Here in this research, The Hierarchical Classifier System(HCS) has been chosen as the mechanism to set these parts. The basic idea of its realization is that SFR and VFR sets are located in the HCS as illustrated in Figure 5. To implement this idea more easily in the computational system, the authors adopt the policy in order to develop the system by the Pitt approach, which would be explained in next.

## 2.2 Classifier System

The SFR and VFR sets are processed by the CS. Solution-Finding applying CS has been performed in various research areas over past two decades. The general type of CS architecture is distinguished by two policies, one is the Michigan approach which contains the delicate apportionment of credit (AOC) system and by means of which it evaluates each single rule as an individual, and the other is the Pitt approach which evaluates the rule sets as an individual without using the AOC system. The former originates from Holland's System, CS-1[6], and the latter, from Smith's system, LS-1[7]. Because the Pitt approach avoids the

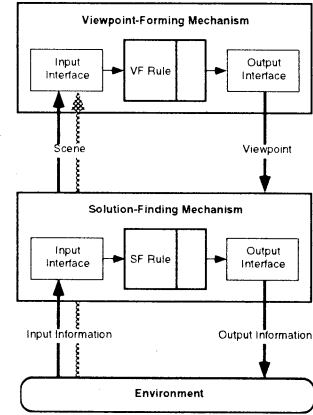


Figure 5: Basic idea of Viewpoint-Forming Mechanism

need for the delicate AOC system, and is supposed to be relatively easier in letting multiple rule sets be located hierarchically, the authors follow this policy to develop the system.

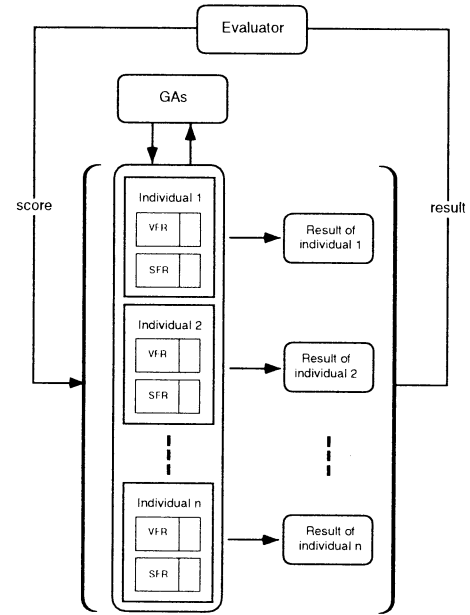


Figure 6: Conceptual overview of system

Figure 6 shows the conceptual overview of the HCS developed by us. One single individual consists of 2 rule sets which are SFR and VFR. Several individuals are prepared, and each individual is evaluated according to the result that it generates. GAs generate new rules by means of genetic operations, that is, crossover and mutation, according to the evaluation score.

As to classifier, a usual definition is previously mentioned in Figure 1. In this research, depending on the type of rule, there exists a subtle difference between classifiers in an individual. In classifiers for

SFR.  $\langle \text{condition} \rangle$  is the criterion with input message and  $\langle \text{action} \rangle$  contains the information for the output message. In classifiers for VFR,  $\langle \text{condition} \rangle$  is the criterion with input message with condition such as in classifiers for SFR, however  $\langle \text{action} \rangle$  contains the information on "how to look at" the input  $\langle \text{message} \rangle$ , in other words, information to determine which property parts the system should compare according to the input  $\langle \text{message} \rangle$  itself.

Hence, the flow of transformation of the input information,  $\langle \text{message} \rangle$ , and matching process is summarized as follows (which corresponds to Figure 7):

1. Compare input  $\langle \text{message} \rangle$  with  $\langle \text{condition} \rangle$  of classifier for VFR: if matched output  $\langle \text{action} \rangle$  for VFR:
2. Determine certain property within  $\langle \text{message} \rangle$  to match with the condition of classifier for SFR according to the  $\langle \text{action} \rangle$  for VFR:
3. Compare  $\langle \text{message} \rangle$  with  $\langle \text{condition} \rangle$  of classifier for SFR partially according to the result of 2: if matched output  $\langle \text{action} \rangle$  for SFR:

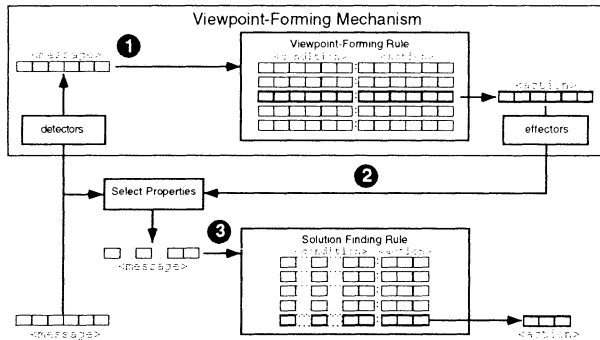


Figure 7: Transformation of  $\langle \text{message} \rangle$  and matching process

### 3 Implementation on the CS and its GAs

The authors developed a prototype system in order to examine the effect of the VFM proposed by us in the previous section. For the implementation of this mechanism, the authors employ the classical path-finding problem as an example. As it is easy to understand its behavior and results of the problem and it is easy to imagine how the number of properties will increase along with the complexity of the experimental space.

#### 3.1 Example Problem: Classical Path-finding

First, the authors explain the setting of the example problem. The authors presume that the system and

the mechanism are not highly specialized and can be applied fairly easily to other problems.

**Problem Setting:** The authors assume several agents in the experimental space which spans  $22 \times 16$  grids. 7 types of Pentomino- like "Building" and 6 types of "Mark" exist in the space (Figure 8). Each grid can contain a building and a mark. Some grid contain only a building and some only a mark. There also exists empty grids in the space where the agent can move. Agents do not know the geometric position of the goal.

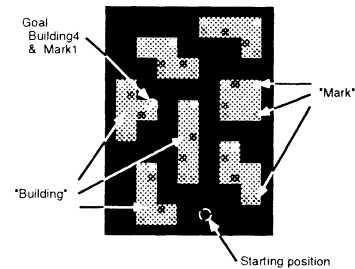


Figure 8: Simulation Space

The starting point is given as a geometric position, and the goal of the agents is given as a combination of a building and a mark. Then, several agents start finding the path to the goal.

**Architecture:** An overall functional diagram of the process for finding the path to the goal is shown in Figure 9.

1. Input the geometric position where agents start finding way to goal: input the combination of a building and a mark which is the goal:
2. Randomly generate a number of rule sets which consists of SFR and VFR classifiers:
3. Send the information around the agent to detectors: translate into  $\langle \text{message} \rangle$  which will be explained in following point:
4. Compare input  $\langle \text{message} \rangle$  with  $\langle \text{condition} \rangle$  of classifier for VFR: if matched output  $\langle \text{action} \rangle$ :
5. Determine a certain property within  $\langle \text{message} \rangle$  to match with the condition of the classifier for SFR according to the  $\langle \text{action} \rangle$  for VFR:
6. Compare  $\langle \text{message} \rangle$  with  $\langle \text{condition} \rangle$  of the classifier for SFR partially according to the result of 5: if matched output  $\langle \text{action} \rangle$ :
7. Send  $\langle \text{action} \rangle$  to effectors: translate into direction of agent movement:
8. Repeat 3. ~ 7. until agent finds the path to the goal or faces deadlock:

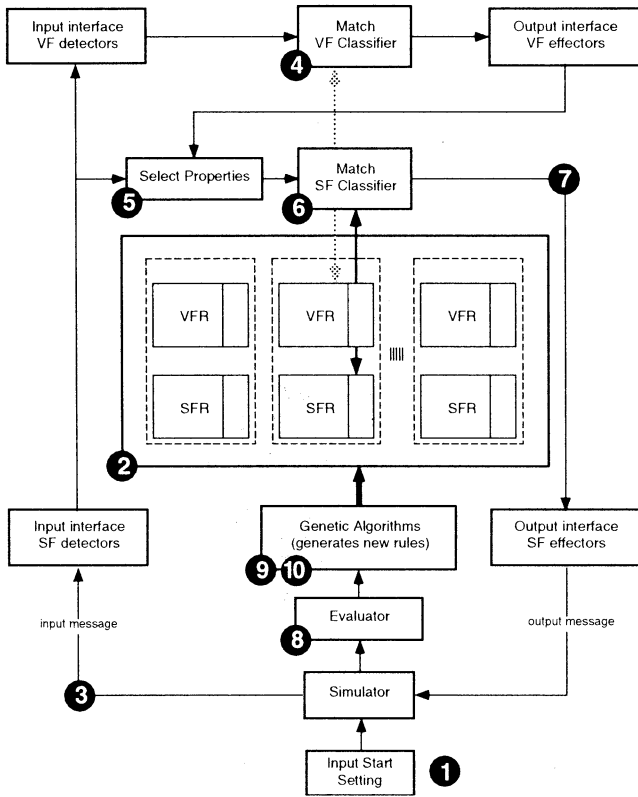


Figure 9: Diagram of prototype system

9. Evaluate rule sets according to the fitness function and make a score of them:
10. Select the rule sets for agents to approach the goal according to the score:
11. Crossover and mutate selected rule sets by genetic operations.
12. Repeat 3. ~ 11. until at least one agent finds the path to the goal which is the certain combination of building and mark.

**Coding:** An agent can see the grid around itself like Wilson's animat[2]. The detector translates the input message around the agent into `<message>`. Also an agent determines the direction in which to move from the output message decoded by effectors from `<action>`. Figure 10 illustrates how to encode the information to `<message>` and how to decode `<action>` to agents movement. In the system, all the properties are based on 3 bits as a single property for the purpose of convenience.

**Fitness Function:** Figure 11 shows the simple fitness function for evaluating the rule sets for the selection of GAs.

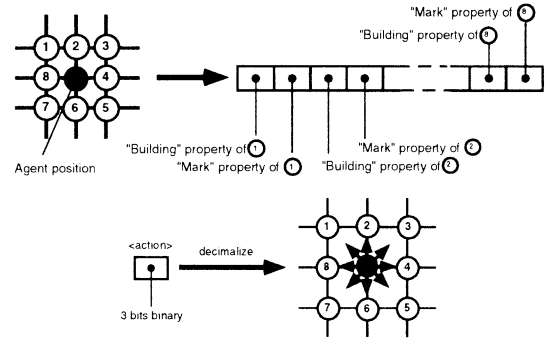


Figure 10: Encoding `<message>` and decoding `<action>`

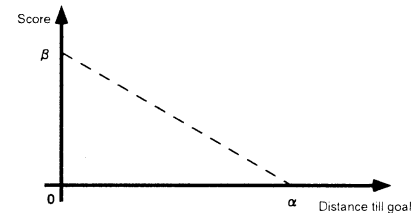


Figure 11: Fitness Function of the System

Evaluator calculates the distance from the last position of the agent to goal position at every session. According to the fitness function above, fixes the score of those rule sets. Note that the problem that system knows the answer to the geometric position of the goal does not occur since this experiment is not the examination of the development of the autonomous agent.

**Genetic Algorithm:** The authors adopt the policy to leaving plenty of offsprings and the inheritance of parents by the next generation in the genetic algorithm shown in Figure 12. The authors allow the system to form various VFR rule sets corresponding to the SFR, which the authors expect will introduce the possibility searching for better combinations of SFR and VFR sets in the system. Mortal individuals are selected according to the reciprocal number of their score and by the use of a simple roulette model.

For comparison purpose, when the authors do not use the VFM, the policy to leave offspring is as shown in Figure 13. There, the number of children is the same as that in the case of using the VFM. In order to make the conditions more similar, the site is different in each crossover to generate the offspring.

**Experimental Parameters:** The authors carry out the experiment using these experimental parameters shown in Table 1. The start position and the goal combination of building and mark are also shown in Figure 8.

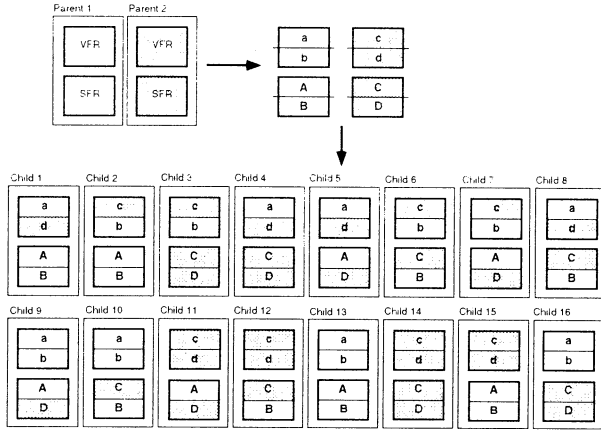


Figure 12: Shift policy for VFR

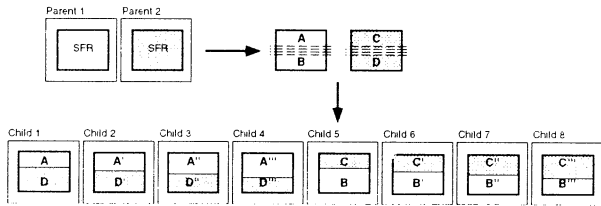


Figure 13: Shift policy for comparison with only SFR

	SFR	VFR
Number of rules in single individual	300	300
Crossover probability	0.5	0.5
Mutation probability	0.05	0.05
String length	51	64
Common parameter		
Number of individuals		200
Value $\alpha$ in Fitness Function		20
Value $\beta$ in Fitness Function		50

Table 1: GA parameters for the experiments

### 3.2 Result

Figure 14 shows the average of the shortest distance from the agent position to goal position until 1000 generations, which is the average of 10 experiments under the conditions mentioned previously. In the Figure 14, the bold line shows the result obtained using the VFM and the narrow line shows that obtained without the VFM. Each experiment uses different random seeds. In this example problem, the assumption of VFR contributes to increasing an effectiveness of system performance.

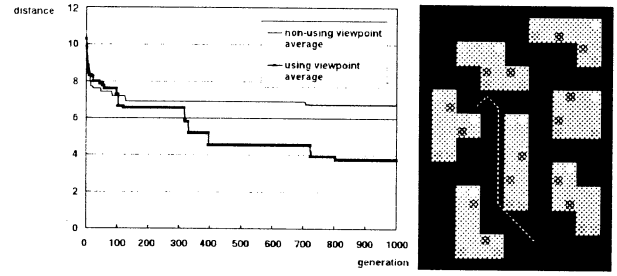


Figure 14: Example result of path-finding

## 4 Conclusions

With the aim of designing a Machine Learning System capable of being extended for application to various fields where a long if-part in the production rule is required, the authors propose a comparatively primitive method. Viewpoint-Forming Mechanism (VFM) in order to realize the system. Using the VFM, the system compares certain properties in if-part of rule sets when matching with the input message. A prototype was developed to evaluate the utilities of the VFM employing the hierarchical classifier system. A simple example problem was used to implement the VFM. In the example problem, the assumption of VFR contributes to increasing an effectiveness of system performance. Though this good result may not easily expanded and insist on the generality of the Viewpoint-Forming Process, it can be said that this framework has fairly good potential.

In our future work, the authors are planning to apply this mechanism to practical problems such as computer-aided designing involving human thinking-process in which the potential of the VFM is utilized.

## References

- [1] J.H. Holland. *Adaptation in Natural and Artificial Systems*. MIT Press, 1975.
- [2] S.W. Wilson. Knowledge growth in an artificial animal. In *Proceedings of an International Conference on Genetic Algorithms and Their Applications*, pages 16-23, 1985.
- [3] M.R. Hilliard, G.E. Liepins, M. Palmer, M. Morrow, and J. Richardson. A classifier-based system for discovering scheduling heuristics. In *Proceedings of the second international Conference on Genetic Algorithms*, pages 231-235, 1987.
- [4] D.E. Goldberg. *Genetic Algorithm in Search, Optimization, and Machine Learning*. MIT Press, 1989.
- [5] J.H. Holland. *Hidden Order: How Adaptation Builds Complexity*. Addison-Wesley, 1995.
- [6] J. Holland and J. Reitman. Cognitive systems based on adaptive algorithm. In *Pattern Directed Inference Systems*. Academic Press, New York, 1978.
- [7] S.F. Smith. *A Learning System Based on Genetic Algorithm*. Ph.D. dissertation, University of Pittsburgh, 1980.

## An Evolutionary Architecture for a Humanoid Robot

Peter Nordin and Mats G. Nordahl  
Chalmers University of Technology  
Institute of Physical Resource Theory  
S-412 96 Göteborg, Sweden  
Email: nordin,tfemn@fy.chalmers.se

### Abstract

This document describes the hardware and software architecture behind the ELVIS robot. ELVIS is bipedal robot with human-like geometry and motion capabilities — a humanoid. ELVIS is also the first robot in a series of planned humanoid experiments, all of which will be primarily controlled by evolutionary adaptive methods. The final goal of our research project is to build a human-sized robot based on a plastic human skeleton to ensure geometric authenticity.

### 1 Introduction

The field of autonomous mobile robotics attracts an accelerating interest. Application areas are plentiful in both industry and academia, but an autonomous mobile robot system also demands high performance of both mechanical components and control software. The many degrees of freedom in a light mobile robot create new problem spaces in control and navigation where conventional methods often fall short. A relatively new and promising area for control of autonomous agents is *evolutionary algorithms*, which are inspired by the main adaptation method in nature — *natural selection*. Most challenging of all autonomous robots are robots that move using legs instead of wheels. Walking robots have very large potential in environments created for humans as well as in more natural terrain. The largest potential is associated with robots of human-like dimensions walking on two legs — *humanoid robots*. Man is the standard for almost all interactions in our world where most environments, tools and machines are adapted to the abilities, motion capabilities and geometry of humans. However, a bipedal humanoid robot demands extreme performance in everything from power supply to computer capacity and control algorithms. But if we succeed in building humanoids, it could be more efficient

to control various machines by these robots than to rebuild all machines for direct computer control. It has been argued that humanoid robots could be the next dominating mechanical industry, as large as or larger than the auto industry. In this paper we briefly describe an evolutionary control architecture that will be the basis for several humanoid robotics experiments.

### 2 The ELVIS Humanoid

ELVIS is a scale model with a height of about 60 cm, built with 42 servos giving a high degree of freedom in legs, arms and hands. The robot will be guided by microphones, cameras and touch sensors. The imminent goals are to walk upright and to navigate through vision. Seven onboard micro-controllers control the servos and sensors. ELVIS is not yet fully autonomous, but the plan includes onboard power supply and main processing unit. The current status is that the robot is assembled with servos and controllers, and that software development has been carried out in parallel with the construction process.

#### 2.1 Software Architecture

The software architecture is built mainly on evolutionary algorithms and specifically Genetic Programming. Evolution is thus used to induce programs, functions and symbolic rules for all levels of control [1, 2]. Three hierarchical layers are used for control:

- Reactive Layer
- Model Building Layer
- Reasoning Layer

##### 2.1.1 Reactive Layer

The first layer is a reactive layer based on on-line evolution of machine code. This method assumes that all

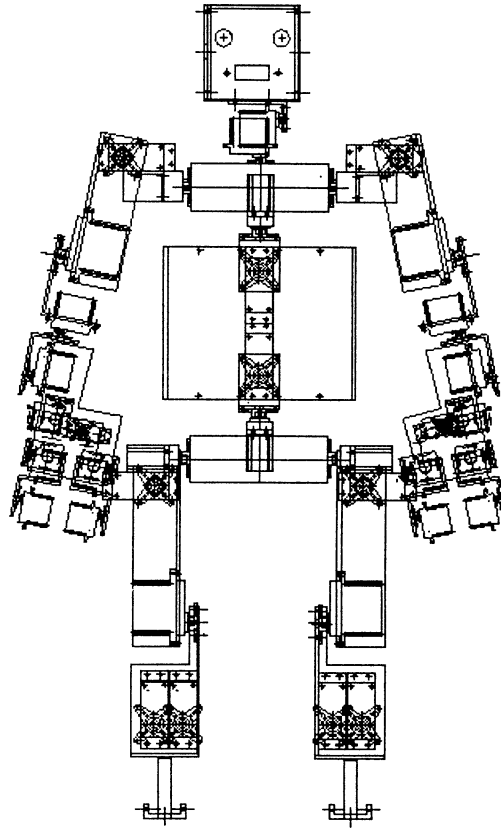


Figure 1: The Elvis Humanoid Robot

fitness feedback is obtained directly from the actual robot. The disadvantage is that the GP individuals spend most of their time waiting for feedback from the physical environment. This results in moderate learning speed, and the constant movement shortens the life-span of the hardware. The benefit of the method is its simplicity, and that the only constraints needed for the models being learned are that they should fulfil their task as a black box. This layer is used for reactive behaviors such as balancing.

### 2.1.2 Model Building Layer

To achieve higher learning speeds and more generic behavior there is a second control layer that works with memories of past events. In this genetic reinforcement learning framework, the system tries to evolve a *model* of the underlying hardware system and problem. The model maps sensor inputs and actions to a predicted goodness or fitness value. The currently best model is then used to decide what action results in optimal pre-

dicted fitness given current sensor inputs. This layer allows the genetic programming system to run at full speed without having to wait for feedback from the environment; instead it fits the programs to memories of past events. The machine code genetic programming approach used is called Automatic Induction of Machine Code GP (AIMGP) [3]. AIMGP is about 40 times faster than conventional GP systems due to the absence of any interpreting steps. In addition, the system is compact, which is beneficial when working onboard a real robot. The model building layer is also used for basic control tasks.

### 2.1.3 Reasoning Layer

The third layer is a symbolic processing layer for higher "brain functions" requiring reasoning. The objective of this layer is to handle high level tasks such as navigation, safety, and energy supply. This layer is built on "genetic reasoning", a method where evolution is used as an inference engine, requiring less



heuristics to guide the inference procedure [3].

Each of these layers consists of modules for various tasks such as balancing, walking and image processing. Some system functions are represented as several modules spanning different layers.

### 3 Experiments

Several experiments are being performed in parallel on ELVIS:

- Balancing
- Walking
- Vision
- Navigation
- Audio orientation
- Manipulation

#### 3.1 Balancing

In this experiment, the robot is set up to learn balancing. The sensory inputs used to learn to balance are touch sensors and two electronic gyros. The actuators are a subset of the more than 40 servos controlling the robot. In the first initial experiments, the robot is suspended in a safety harness, that prevents it from falling to the ground if it loses its balance. If this happens, a sensor in the safety lines is activated, and a servo lifts the robot and returns it to an upright position. Experiments on balance are performed both in the reactive and model building layers. The model building layer puts less stress on the mechanical parts of the robot by checking hypotheses mainly against data in the memory vectors, which results in simple smooth behavior. On the other hand, it is a less complex task to get the system to learn in the purely reactive layer. The fitness or goodness criterion of the experiments is a combination of three inputs:

1. Inputs from pressure sensors on the feet giving information on the center of gravity.
2. Input from the two gyros in the top of the head trying to minimize head movements.
3. Information from the safety line sensor indicating total loss of balance.

The robot is provoked by disturbances of adjustable size obtained from random arm movements and wind pressure from a rotating fan.

#### 3.2 Walking

The walking experiments are performed in the reactive and model building layer. Two different approaches are used:

- A simple setup where the goal of the robot is to move forward without constraints as long as it maintains balance. This experiment is conducted using the reactive layer.
- Another set of experiments solves the problem by dividing the task into subtasks — first shifting from two legs to one, then shifting the weight from one foot to the other, and finally dividing weight on both feet. These experiments are performed both in the reactive and model building layers.

The controlling hardware that has been constructed can measure the power consumption of each servo individually, and we plan to incorporate efficiency in the fitness of the walking model. This will potentially also give a smoother, more natural way of moving.

#### 3.3 Vision

The vision experiments are aimed at creating a 3-D model of the environment. The hardware of the system uses two CCD cameras for stereo vision. In the initial phase, the following two experiments are being performed:

- First we evolve a program which produces a 3-D map of the environment directly from the pixels of the two cameras. Using the pixels directly gives a higher potential for efficient execution since the evolutionary system is free to evolve any heuristic filter it may need, instead of forcing a set of pre-defined heavy computations on the system. The price one has to pay is that it is harder for the evolutionary system to find all information in the raw data.
- The second experiment uses the 3-D map from the first experiment to evolve representations of 3-D *objects*. The genetic system tries to generalize from the 3-D map to a list of 3-D geometric shapes such as boxes, cones and spheres. This results in more complete hypothesis that includes extrapolation of hidden surfaces.

#### 3.4 Navigation

The goal of the navigation module is to integrate the modules described above, and to give the robot the

ability to walk in an office following walls and avoiding obstacles. Navigation also includes planning of the path to follow in order to arrive at a certain point. The navigation task is mostly achieved in the third symbolic reasoning layer, using information from vision, balancing and walking. Genetic Reasoning [3] is used to evolve plans and prove that they fulfill the goals to an acceptable level.

### 3.5 Audio orientation

The robot has two microphones in the head for stereophonic hearing. At present, these are mainly used for focus of attention. A GP system is used to evolve a function or program, which can give the direction of a sound. Future experiments include separation of sound sources and recognition of commands.

### 3.6 Manipulation

ELVIS has arms and hands with three fingers each including a highly maneuverable thumb. The fingers are equipped with touch sensors reacting to pressure from 10 g up to tens of kilograms. The manipulation experiments take place in the symbolic layer and integrate movement, touch sensors and vision to isolate an object and pick it up with one hand and then moving it to the other hand.

## 4 Current status

ELVIS has been now assembled, and is complete with the exception of CCD-cameras and fingers. The software architecture has been evaluated on off-line experiments. Initial experiments in simulations confirm the feasibility of the method, but major evaluations using the robot are still under way. ELVIS is not yet autonomous. It is so far controlled by remote computers and has a remote power supply. The aim is to have the humanoid controlled by a handheld PC running NT or possibly LINUX. Space is also reserved for battery packs. The total weight of the autonomous version of ELVIS is expected to be less than 5 kg.

## 5 Summary and Conclusions

The ELVIS humanoid robot is a very complex system both as far as hardware and software are concerned. It is unique in many ways: size, weight, degrees of freedom, possibility of autonomy and control method.

Nature has shown that evolution is a very powerful tool for controlling complex systems in an adaptive way. Our hypothesis is that evolutionary systems such as Genetic Programming and AIMGP are very well suited for control of complicated systems. We have chosen to build the control architecture almost exclusively on EAs operating on a wide variety of tasks and using several different methods for evolution and representation. The experiments are still at an initial stage, but we believe that the hardware and software architecture may be of interest to the research community in EAs, robotics and control. We will use the experiences from ELVIS for the construction of a full-size humanoid robot built on a plastic human skeleton.

## Acknowledgements

Peter Nordin gratefully acknowledges support from the Swedish Research Council for Engineering Sciences. Special thanks to Manne Kihlman and Marcus Tallhamn, who did excellent work in developing the hardware design for ELVIS. Per Svensson, Björn Andersson, Rikard Karlsson, Thorbjörn Engdahl, Anders Eriksson and Christopher Graae have also made significant contributions to various parts of the project.

## References

- [1] Banzhaf, W., Nordin, P., Keller, R. E., and Francone, F. D. (1997). *Genetic Programming – An Introduction. On the automatic evolution of computer programs and its applications*. Morgan Kaufmann, San Francisco, and d-punkt, Heidelberg.
- [2] Koza, J. R. (1992). *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA.
- [3] Nordin, J.P. (1997), *Evolutionary Program Induction of Binary Machine Code and its Application*. Krehl Verlag, Muenster, Germany

## A Development of Computer Aided Identification for Systems Under Control

Kunihiko OURA	Takayuki MURAKOSHI	Kageo AKIZUKI	Izumi HANAZAKI
Waseda Univ.	Waseda Univ.	Waseda Univ.	Tokyo Denki Univ.
3-4-1, Ohkubo	3-4-1, Ohkubo	3-4-1, Ohkubo	Ishizaka, Hatoyama
Shinjuku-ku, Tokyo	Shinjuku-ku, Tokyo	Shinjuku-ku, Tokyo	Hiki-gun, Saitama
169-8555, JAPAN	169-8555, JAPAN	169-8555, JAPAN	350-0394, JAPAN

### Abstract

This paper treats practical aspects of system identification, that a development of computer aided identification under control is considered. There have been many available tools for system identification frequently developed these days, and their application is normally aimed at experts and relies on the experience and cleverness of the user. But most of the tools were not developed assuming the cases that the data are collected by closed-loop experiment, that is, under control. The authors have studied the subject for several years, and it is time to propose our techniques in userfriendly manner to engineers. In this paper, they develop the supporting system for identification of closed-loop system based on their works. Simulation examples whose data are observed by practical plant, will show usefulness of the supporting system.

## 1 Introduction

The importance of identification in closed-loop operation, or identification of the system under control, has been enhanced in the recent years [1], [2]. There are many works on the subject, but most of their topic is on the identifiability of closed-loop system [3] or design method of controller considering the closed-loop property [4]. They did not refer to the estimation method of the true plant straightforward. Then the identification technique to estimate an exact model from measurement data has been expected for a long time.

The authors have been studying the subject for several years. They adopted a linear model and investigated the estimation method of model structures, especially orders, delay-time of the plant. Assuming that the identifiability of the closed-loop system is guaran-

teed, some specific techniques [5] [6] were developed. Then we should propose our techniques in userfriendly manner to engineers who are working in identification for practical plants.

This paper treats such practical aspects, that a development of computer aided system identification under control is considered. There have been many available tools for system identification frequently developed these days, and their application is normally aimed at experts and relies on the experience and cleverness of the user. Many tools have been programmed in MATLAB<sup>TM</sup> [7], a numerical computation software, so the authors also developed their system under MATLAB<sup>TM</sup>. Simulation example whose data are generated by practical plant, will show usefulness of their system.

## 2 Problem statement

A set of input and output data are collected from the system depicted in fig.1. The system is composed of plant and controller, where the plant is described by linear, time-invariant, discrete-time model with additive noise (1), and controller designed by regulation is given by (2).

$$A(z^{-1})y(k) = z^{-d}B(z^{-1}) + w(k) \quad (1)$$

$$F(z^{-1})u(k) = G(z^{-1})y(k) \quad (2)$$

It is assumed that the system satisfies identifiability conditions so that unbiased and consistent estimates are given if we have the true model structure. The purpose is to develop the identification procedure of selecting a suitable model, and build a supporting system for them.

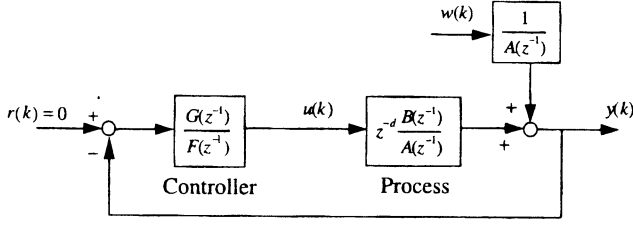


Figure 1: Block diagram of the system

### 3 Model selection for closed-loop data

It is often necessary in practice to identify a process with input and output data collected from closed-loop system. All criteria for model selection can not be valid for such systems, because there exists correlation between input and output. The authors considered an estimation method of orders, delay-time from input and output, and developed an estimation procedure. They are shown as follows:

#### step1: evaluation of additive noises

In the first step, evaluation of the noises added to the system is needed. Comparison of the power spectrum of input and output with the controller in frequency domain enables them. If it is estimated that the noise has enough power to the input part, orders and delay-time can easily be estimated by criteria for open-loop systems, that is prediction errors, AIC, FPE, etc. Otherwise we have to proceed to the next steps.

#### step2: choice of candidates for DARX model

In this step, candidates of orders for DARX model (The plant : (1)) is chosen by careful treatment of ARX or ARMA model. Choice of estimated model depends on the power of additive noise to the input. If there exists additive noise with less power to the input part, we should use ARX model (3), and we should use ARMA model (4) when it is evaluated as noise-free. Therefore, we should consider two cases below.

$$\hat{A}(z^{-1})y(k) = \hat{B}(z^{-1})u(k) + w(k) \quad (3)$$

$$\hat{A}(z^{-1})y(k) = \hat{B}(z^{-1})w(k) \quad (4)$$

#### case1: the case when small input noise exists

The system is described by (5) from (1) and (2) in this case, so that ARX model is fitted to the data for some candidates of orders.

$$\begin{aligned} & \{L(z^{-1})A(z^{-1}) + M(z^{-1})G(z^{-1})\}y(k) \\ &= \{z^{-d}L(z^{-1})B(z^{-1}) - M(z^{-1})F(z^{-1})\}u(k) \\ &+ L(z^{-1})w(k) + M(z^{-1})F(z^{-1})v(k) \end{aligned} \quad (5)$$

We should choose the appropriate ARX model orders  $(\hat{n}_a, \hat{n}_b)$  which satisfy (6) and (7), where  $(\hat{n}_l, \hat{n}_m)$  stands for arbitrary real-coefficient polynomials.

$$\hat{n}_a = \max\{n_a + n_l, n_g + n_m\} \quad (6)$$

$$\hat{n}_b = \max\{n_b + d + n_l, n_f + n_m\} \quad (7)$$

We suppose  $\hat{n}_a = \hat{n}_b$  to avoid increasing the number of pairs  $(\hat{n}_a, \hat{n}_b)$ , and calculate prediction errors for all ARX models. By evaluating the prediction errors and spectrum carefully, candidates of model orders are chosen, denoted as  $\hat{n}$ . Considering the fact that delay-time is large compared with the orders in this kind of control systems, we have (8).

$$\hat{n} = n_b + d + n_l \quad (8)$$

Here, we must take care of numerical accuracy of least square method (LSM). If  $\hat{n}$  is large, data-matrix will closed to be singular. It can be checked by condition numbers of data-matrix, so we should also check them in this step.

#### case2: the case when input is noise-free

In this case, the system is described by (9) from (1) and (2), so that ARMA model is fitted to the data for several orders.

$$\begin{aligned} & \{A(z^{-1})F(z^{-1}) + z^{-d}B(z^{-1})G(z^{-1})\}y(k) \\ &= F(z^{-1})w(k) \end{aligned} \quad (9)$$

We assume  $\hat{B}(z^{-1}) = F(z^{-1})$  and calculate prediction errors for all ARMA models, then we have candidates ARMA model orders, denoted as  $\hat{n}$ . By using the same assumptions as the case1, we have (10).

$$\hat{n} = n_b + d + n_d \quad (10)$$

As we see, the order  $\hat{n}$  is chosen in both cases.

#### step3: selection of orders, delay-time

Estimates  $\hat{A}(z^{-1})$  of order  $\hat{n}$  for each case should be checked first. It is expected that order  $\hat{n}$  can be reduced to  $\hat{n}_a$  by overparametrization, then candidates of  $\hat{n}_b$  and  $\hat{d}$  are also chosen. If we denote the reduced order of  $\hat{A}(z^{-1})$  as  $\hat{m}$ , it is described as (11) and (12) for each case from (3), (4), (5) and (9).

$$\text{Case1: } \hat{m} = \max\{n_a + n_l, n_g + n_m\} \quad (11)$$

$$\text{Case2: } \hat{m} = n_a + n_f \quad (12)$$

Candidates of the orders and delay-time of DARX model are given as follows, by using  $\hat{n}$  and  $\hat{m}$ .

$$0 < \hat{n}_a \leq \hat{m} \quad , \quad \hat{n}_a \leq \hat{n}_b \quad , \quad \hat{n} - \hat{m} \leq \hat{d} \leq \hat{n}$$

The first equation turns to  $\hat{n}_a = \hat{m}$  in case2. Orders and delay-time can be chosen among the candidates by checking prediction errors, whiteness of the errors and pole-zero placement.

From the above steps in consideration, we have an identification procedure for closed-loop system depicted in fig2.

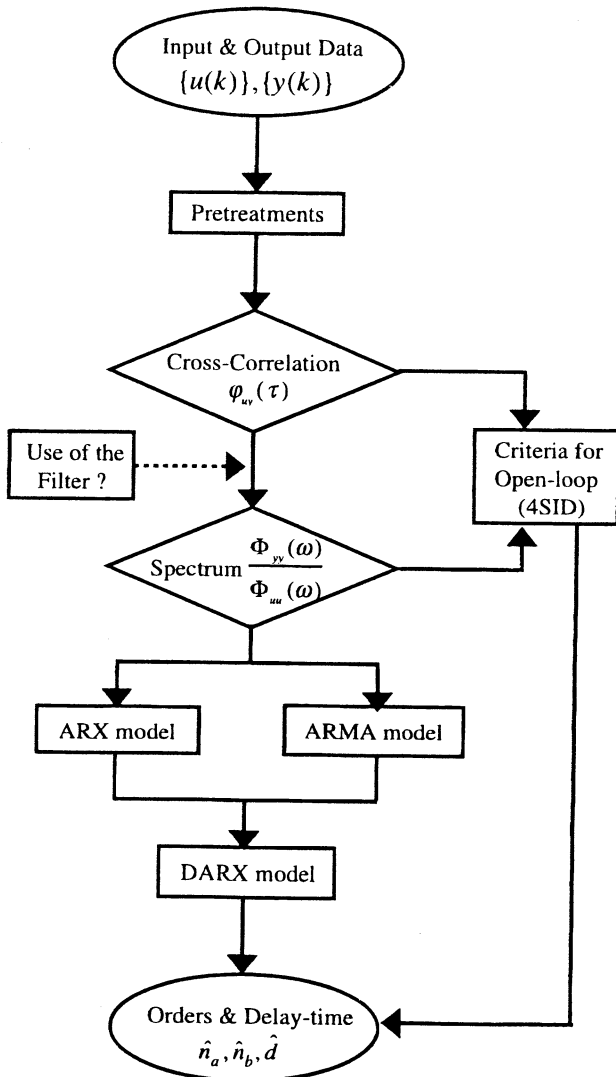


Figure 2: System identification procedure for closed-loop system

#### 4 Development of supporting system

We developed the supporting system for the plant, consisted of the procedure considered in the preceding section. Here, MATLAB<sup>TM</sup> is used to construct the system in userfriendly manner. In order to build a useful system, dialogical method is adopted. Here, some parts of our system are introduced in pictures.

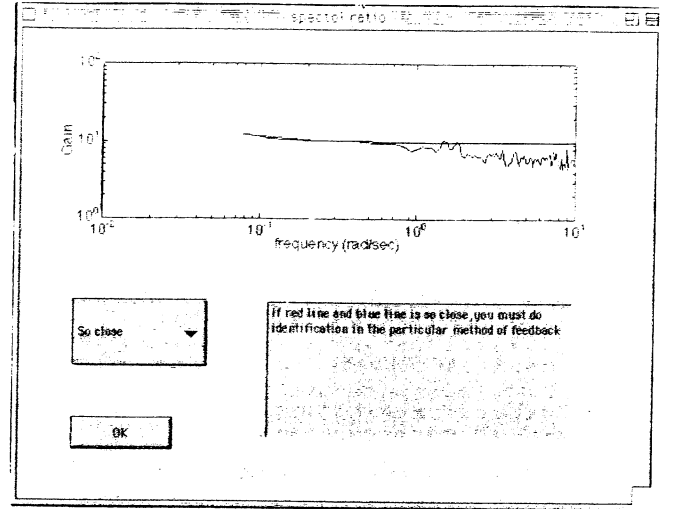


Figure 3: Comparison of the power spectrum

Fig.3 shows properties of the data in frequency domain. The user must evaluate their properties, that is the difference between them. If he recognizes that two lines are so close, he should click "OK". If not, he should find another step from the pulldown menu.

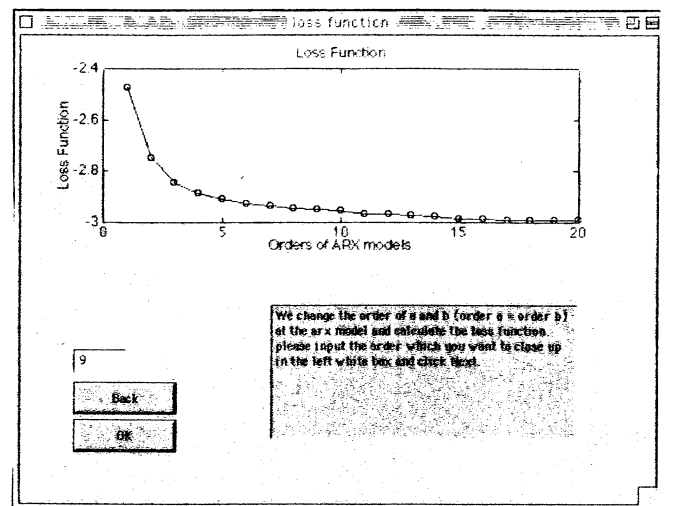


Figure 4: Prediction errors for ARX models

Fig.4 shows the prediction errors for ARX models. If the user cannot find the specific point, he can change the range of orders by entering the number (shown as "9"). Then he has fig.5.

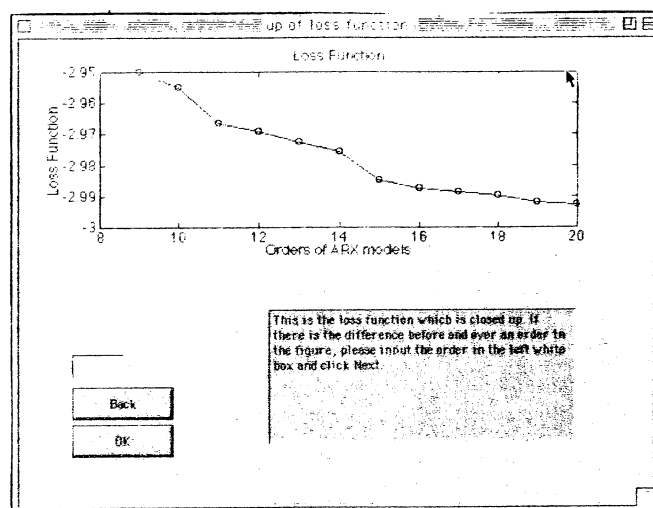


Figure 5: Prediction errors for ARX models

The user must pursue the procedure backward when he recognizes that he made an improper decision. So that "back" button is also important to have appropriate results. Here, we only illustrate some pictures of the system, but the estimation procedures will be displayed in detail by the system at the conference presentation.

## 5 Summary

We have considered a computer aided identification for systems under control. It consisted of the estimation procedure which was developed by authors, and aims for proper estimation of orders, delay-time from input-output data. Supporting system works dialogically, and simulation examples will show usefulness of the estimation procedure and developed supporting system.

## References

- [1] V.Hof and R.Schrama, "Identification and Control - Closed-loop Issues" *Automatica*, Vol.38, No.12, pp.1751-1770,1995.
- [2] L.Ljung, *System Identification -Theory for the user*, Prentice Hall, 1987.
- [3] I.Gustavsson,L.Ljung and T.Soderstrom "Survey Paper" *Automatica*, Vol.13, pp.59-75,1977.
- [4] R.Schrama, "Accurate Identification for Control, the Necessity of an Iterative Scheme" *IEEE Trans.*, Vol.AC-37, No.7, pp.991-994,1992.
- [5] K.Oura,I.Hanazaki and K.Akizuki, "Estimation of Orders and Delay-time of the Process in Closed-Loop Control System", *Trans. IEE of Japan*, Vol.117-C, No.2, pp.128-135,1997.
- [6] K.Oura,I.Hanazaki and K.Akizuki, "Identification of the Process Operating in Closed-Loop System Using Filtered Data", *Proc. of 27th SSS*, pp.25-30,1995.
- [7] L.Ljung, *System Identification Toolbox User's Guide*, The Math Works Inc., 1995.
- [8] S.Korner and R.Schumann, "A New Framework for Computer Aided Identification in the Process Industry", *Proc. of MIC'98*, pp.437-439,1998.

## ATR's Artificial Brain (CAM-Brain) Project: A Sample of What Individual CoDi-1Bit Model Evolved Neural Net Modules Can Do

Hugo de Garis<sup>(1)</sup>, Michael Korkin<sup>(2)</sup>, Felix Gers<sup>(3)</sup>, Norberto Eiji Nawa<sup>(1)</sup>, Michael Hough<sup>(4)</sup>,

(1) Dept. 6, ATR-HIP, Kyoto, Japan, <http://www.hip.atr.co.jp/~{degaris,xnawa}>

(2) Genobyte Inc., Boulder, Colorado, USA, <http://www.genobyte.com>

(3) Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Lugano, Switzerland, <http://www.idsia.ch/~felix>

(4) Computer Science Department, Stanford University, Stanford, USA, <http://www.stanford.edu/~mhough>

### Abstract

This paper presents a sample of what evolved neural net circuit modules using the so called CoDi-1Bit neural network model can do. The concern is to study the important issue of evolvability of the cellular automata (CA) based neural network circuits which grow and evolve in special FPGA (Field Programmable Gate Array) hardware. The specialized hardware which performs this evolution is labeled the CAM-Brain Machine (CBM). The CBM should make practical the assemblage of 10,000s of evolved neural net modules into humanly defined artificial brains. A software simulator of the CBM has been used to see just how evolvable and functional individual evolved modules can be. This paper reports on some of the results of these simulations.

**Keywords:** evolutionary neural networks, hardware implementation, CAM-Brain project, CoDi neural model

### 1 Introduction

ATR's CAM-Brain Project [1] aims to build a large-scale brain-like neural network system in hardware. The essential ingredient in this project is a special piece of hardware, based on Xilinx XC6264 FPGAs which grow and evolve cellular automata based neural network circuits (modules) at electronic speeds. This machine, called CAM-Brain Machine (CBM), can update the cellular automata (CA) cells which form the basis of the neural network at a rate of 150 billion a second. Although under this framework to speed up evolution is unquestionably a necessary requirement, there is no assurance whatsoever that all sorts of modules can be evolved only with a great increase in computation power. There may be other factors that allow or enhance evolution. In order to investigate more thoroughly the issue of module evolvability, i.e. what sort of modules can be evolved and how, some preliminary experiments were performed, and the results are presented in this paper.

The remainder of this paper is structured as follows. Section 2 gives a brief description of the "CoDi-1Bit" model, which is implemented by the CAM-Brain Machine (CBM), whose electronic restrictions impose a 1 bit neural signaling model. Section 3 presents a sample of evolved CoDi modules, showing their functionalities and evolvabilities. Section 4 summarizes.

### 2 The CoDi-1Bit Neural Network Model and CAM-Brain Machine (CBM)

The CAM-Brain Machine (CBM) [4] implements a so called "CoDi" (i.e. Collect and Distribute) [2] neural model. It is a cellular automata based neural network model especially developed to make neural network functioning much simpler and more compact compared to the original neural model [1] and enable its implementation in dedicated hardware tools.

In order to evolve one neural module, a population of 30-100 modules is run through a genetic algorithm for 200-600 generations, resulting in up to 60,000 different module evaluations. Each module evaluation consists of, first, growing a new set of axonic and dendritic trees in a 3D cellular automata (CA) space, guided by the module's chromosome, which encodes the growth instructions of each of the cells in the space. The final product of the growth phase of each module is a set of tree-like structures that interconnect the neurons through axonic and dendritic connections. Evaluation is continued by sending spiketrains to the module through its efferent axons (external connections) to evaluate its performance (fitness) by looking at the outgoing spiketrains. (For more details on the CoDi neural model and the CBM, see [2, 4].)

### 3 A Sampler of CoDi-1Bit Evolved Neural Net Modules

In order to investigate more thoroughly the issue of evolvability of neural network modules under the





Evolved ctd.

The fitness definition was similar to the above. If 0 in the first (0) block, 12 points, if 1 in the second (1) block, 7 points, if 0 in the third block (0), 3 points, if 1 in the fourth block (1), 2 points, if 0 in the fifth block (0), 1 point. Hence a perfect score would be  $30 * 12 + 20 * 7 + 24 * 3 + 16 * 2 + 20 * 1 = 624$ . These weightings were chosen so as to encourage the earlier outputs to be correct before the later outputs. Population size was 30, no crossover. This result converged after about 100 generations with a fitness value of 0.957.

### 3.4 Switchable Dual Function Module

More specifically, two fixed position input points IN and SWITCH were placed at positions (8,8,0) and (16,16,0) for a rectanguloid of 24\*24\*18 3D CA cells, with a fixed output point at position (11,12,9). If the output point was not an axon, fitness was defined to be zero.

The bitstrings below show the outputs for the two cases, firstly with SWITCH off, then on. Over 90 clock ticks, the first output had 42 more 1's than the second output.

[illegible]

$$IF(S_1 > S_2)$$

$$IF(S_1 < S_2)$$

The term  $0.001 * (S_1 + S_2)$  was used to encourage circuits to give nonzero output at the output point. The terms  $100 * (S_2 - S_1)$  and  $10000 * (S_1 - S_2)$  encouraged differences in the two outputs, with a strong preference for the first case to give more 1's in the output.

### 3.5 Pattern Detector Module

Square wave input 111000111000...

Since the CoDi modules seem capable of evolving such detectors, it may be possible to evolve modules which are capable of detecting a specific phoneme analog input, e.g. the spike train (bitstring) that represents the time dependent analog signal. In a manner similar to the above, one could input the signal in the first experiment, and a random signal in the second,

### 3.6 Hubel-Wiesel Line Motion Detector Module

### Line Motion Input Case

```
000000000000000000000000010011010011011111111111
11111111111111111111111111111111111111111111111
```

### Output

```
00000000000000000000000000000001000000000000000010
0010101010101010101010101010101010101010101010101
```

is that one loses scientific understanding, due to the overwhelming structural and dynamical complexity of these CoDi circuits.

## 4 Summary and Conclusions

The issue of evolvability is always an open question, because not all modules evolve equally well. Criteria for good evolvability are not well understood. However, in practice, if it is found that a particular module does not evolve well, then alternative modules with different functional specifications can often be found to solve the same problem and that these alternative modules do evolve well.

## References

- [1] Hugo de Garis. An artificial brain : ATR's cam-brain project aims to build/evolve an artificial brain with a million neural net modules inside a trillion cell cellular automata machine. *New Generation Computing Journal*, 12(2), July 1994.
- [2] Felix Gers, Hugo de Garis, and Michael Korkin. Codi-Bit: A simplified cellular automata based neuron model. In *Proceedings of AE97, Artificial Evolution Conference*, October 1997.
- [3] Michael Hough, Hugo de Garis, Michael Korkin, Felix Gers, and Norberto Eiji Nawa. Spiker: Analog waveform to digital spiketrain conversion in ATR's artificial brain (CAM-brain) project. In *Proceedings of the Fourth International Symposium on Artificial Life and Robotics (AROB'99)*, January 1999. In print.
- [4] Michael Korkin, Hugo de Garis, Felix Gers, and Hitoshi Hemmi. CBM (CAM-Brain Machine): A hardware tool which evolves a neural net module in a fraction of a second and runs a million neuron artificial brain in real time. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, July 1997.
- [5] Michael Korkin, Norberto Eiji Nawa, and Hugo de Garis. A 'spike interval information coding' representation for atr's CAM-brain machine (cbm). In *Proceedings of the Second International Conference on Evolvable Systems: From Biology to Hardware (ICES'98)*. Springer-Verlag, September 1998.

## SPIKER: Analog Waveform to Digital Spiketrain Conversion in ATR's Artificial Brain (CAM-Brain) Project

Michael Hough<sup>(1)</sup>, Hugo de Garis<sup>(2)</sup>, Michael Korkin<sup>(3)</sup>, Felix Gers<sup>(4)</sup>, Norberto Eiji Nawa<sup>(2)</sup>

(1) Computer Science Department, Stanford University, Stanford, USA, <http://www.stanford.edu/~mhough>

(2) Dept. 6, ATR-HIP, Kyoto, Japan, <http://www.hip.atr.co.jp/~{degaris,xnawa}>

(3) Genobyte Inc., Boulder, Colorado, USA, <http://www.genobyte.com>

(4) Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Lugano, Switzerland, <http://www.idsia.ch/~felix>

### Abstract

This paper presents an algorithm which converts an arbitrary analog time-varying signal into a digital spiketrain (a bit string of 0's interspersed with 1's), where the information is contained in the spacing between the spikes. This conversion is an important ingredient in the CAM-Brain Project, as it allows the user of ATR's CAM-Brain Machine (CBM) to think entirely in terms of analog signals, and not the more abstract, visually rather meaningless, spike-trains. The SPIKER conversion completes a package of which makes the evolution of individual CBM based neural network modules easier to think about and to accomplish.

**Keywords:** evolutionary neural networks, hardware implementation, CAM-Brain project, CoDi neural model, information representation schemes

## 1 Introduction

ATR's CAM-Brain Project [1] aims to build a large-scale brain-like neural network system in hardware. The essential ingredient in this project is a special piece of hardware, based on Xilinx XC6264 FPGAs which grow and evolve cellular automata based neural network circuits (modules) at electronic speeds. This machine, called CAM-Brain Machine (CBM) [3], can update the cellular automata (CA) cells which form the basis of the neural network at a rate of 150 billion a second, and can complete a full run of a genetic algorithm with tens of thousands of circuit grows and fitness evaluations of those grown circuits in about one second. Tens of thousands (and higher magnitudes) of evolved neural net modules can be evolved and assembled into humanly defined artificial brain architectures. The cellular automata based neural net model used in the CBM had to be simple enough to be implementable in hardware. The constraints imposed

by the electronics were rather severe, thus, the neural model used, called "CoDi" (which stands for "Collect and Distribute, see [2] for details on the model) could not afford to give many bits to the states of the neural signals which traverse the grown neural nets. In fact, the CoDi model has only single bit signaling. Thus, the inputs and outputs of each CoDi module are bit-strings (in this context, spiketrains).

The problem that arises then is how to interpret the meaning of a spiketrain input or output, i.e. to choose a representation for the spiketrains. After some initial experimentation with various representation methods, we eventually settled on one we called Spike Interval Information Coding (SIIC) [4], which convolves the spiketrain output with a digitized analog convolution function, inspired by an hypothesis to explain the neural coding in biological neural networks presented in [6].

The result of this convolution is an analog waveform output (usually time varying) which can then be compared to some user supplied analog target waveform. The fitness of the CoDi module grown and signaled by the CBM is then a function of the sum of the absolute differences between the target and the actual analog waveform values at each clocktick. Experiments were performed using SIIC to generate sine curves and sum of sines and cosines. Constantly firing binary inputs were supplied to a CoDi module, which was evolved to output a spike train which was decoded using the SIIC. The final product of the SIIC would then be compared to the target function. Fig. 1 shows one example of an evolved sinusoidal curve with the actual spikes that generated it.

This paper shows how the opposite operation, i.e. transform an analog waveform into a spiketrain, can be done, with a fairly simple algorithm. The algorithm, named Hough Spiker Algorithm (HSA) will be very useful to evolutionary engineers (EEs), because they will be able to think entirely in terms of ana-

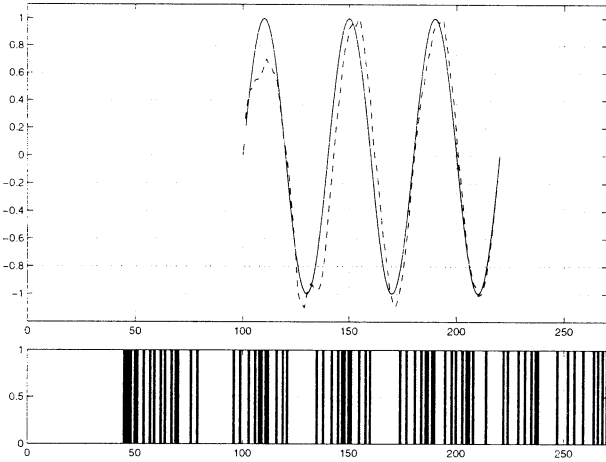


Figure 1: Three periods of a sinusoidal wave generated by the CoDi model and SIIC method. The lower figures show the actual spikes that generated the waveform.

log waveform inputs and outputs when evolving CoDi modules.

The remainder of this paper is structured as follows. Section 2 gives a brief description of the SIIC representation method, which converts a spiketrain into an analog signal. Section 3 presents the Hough Spiker Algorithm (HSA), which does the opposite, i.e. converts an analog signal into a spiketrain, and shows some preliminary results of simulation experiments. Section 4 summarizes and describes the future work.

## 2 Spike Interval Information Coding (SIIC)

The constraints imposed by programmable hardware devices are such that the neural model had to be very simple in order to be implementable within those constraints. Consequently, the signaling states in the CoDi model were made to contain only 1 bit of information (as happens in nature's "binary" spiket trains). The problem then arose as to interpretation. How were we to assign meaning to the binary pulse streams, i.e. the clocked sequences of 0's and 1's, which are neural net module's inputs and outputs? Ultimately, after several experiments were ran with different representation methods [5], a representation which convolves the binary pulse string with a convolution function was chosen. This representation, named Spike Interval Information Coding (SIIC), was inspired by a novel hypothesis of neural coding presented in [6]. The SIIC

representation method delivers a real valued output at each clock tick, thus converting a binary pulse string into an analog time dependent signal. For more details on the method, refer to [4].

## 3 Hough Spiker Algorithm (HSA)

Although the SIIC showed to be efficient in translating the bitstrings output by the CoDi modules into analog waveforms, it only solves half the problem. In order to enable the other way around, i.e., to convert analog waveforms into spiket trains, the Hough Spiker Algorithm (HSA) was developed. This conversion is needed as an interface between the motors/sensors of the robot bodies (e.g. a kitten robot) that the artificial brain controls, and the brain's CoDi modules. Moreover, it is also very useful in the design process of the modules, as it enables one to think entirely in terms of analog signals, in both the inputs and outputs, rather than in an abstract, visually unintelligible spiketrain format. An analog time varying waveform input, processed by the HSA, is converted into an equivalent spiketrain. This spiketrain, if processed by the SIIC, should give back the original analog signal.

In order to give an intuitive feel of the HSA, consider a spiketrain consisting of a single pulse (i.e. all 0's with one 1). When this pulse passes through the convolution function window, it adds each value of the convolution function to the output in turn. At  $t = 0$  its value will be the first value of the convolution filter, at  $t = 1$  its value will be the second value of the convolution filter, etc. Just as a particular spiketrain is a series of spikes with time delays between them, so too the convolved spiketrain will be the sum of the convolution filters, with (possibly) time delays between them. At each clock tick when there is a spike, add the convolution filter to the output. If there is no spike, just shift the time offset and repeat. In graphical form, with a generic spiketrain:

```

1101001      (spiketrain)
1 4 9 5 -2 (convolution filter)
      t -> 0  1  2  3  4  5  6  7  8  9 10
out:
1          1  4  9  5 -2
1              1  4  9  5 -2
0                  0  0  0  0  0
1                      1  4  9  5 -2
0                          0  0  0  0  0
0                              0  0  0  0  0
1                                  1  4  9  5 -2
-----
```

1 5 13 15 7 7 6 2 9 5 -2  
(analog waveform values)

In the HSA deconvolution algorithm, we take advantage of this summation, and in effect do the reverse, i.e. a kind of progressive subtraction of the convolution function. If at a given clock tick, the values of the convolution function are less than the analog values at the corresponding positions, then subtract the convolution function values from the analog values. The justification for this is that for the analog values to be greater than the convolution values, implies that to generate the analog signal values at that clock tick, the CoDi module must have fired at that moment, and this firing contributed the set of convolution values to the analog output. Once one has determined that at that clock tick, there should be a spike, one subtracts the convolution function's values, so that a similar process can be undertaken at the next clock tick. For example:

```
(analog waveform values)
1 5 13 15 7 7 6 2 9 5 -2
1 4 9 5 -2 (convolution filter)
?: conv.filter < analog wave vals., so spike: 1
0 1 4 10 9 7 6 2 9 5 -2
1 4 9 5 -2
?: less, so spike: 11
0 0 0 1 4 9 6 2 9 5 -2
1 4 9 5 -2
?: not less, so no spike: 110
0 0 0 1 4 9 6 2 9 5 -2
1 4 9 5 -2
?: less, so spike: 1101
0 0 0 0 0 1 4 9 5 -2
1 4 9 5 -2
?: not less: 11010
0 0 0 0 0 0 1 4 9 5 -2
1 4 9 5 -2
?: not less: 110100
0 0 0 0 0 0 1 4 9 5 -2
1 4 9 5 -2
?: less, so spike: 1101001 (original spikes)
0 0 0 0 0 0 0 0 0 0
```

It is assumed that spiking will irreversibly raise the value of the convolved output. If the convolution filter value at a given clock tick is less than that of the target waveform, spiking will bring the two values closer together. If the waveform value is still too low after a spike has occurred, a near future spike will bring the two closer together.

Figures 2 and 3 show two examples of the HSA in action. The original input analog signals are shown

as dashed lines. The spiketrain resulting from each analog input is sent into the SIIC convolution process. The resulting analog output should be very close to the original waveform and is shown as a solid line. The third line near the bottom is the absolute difference (error) between the two analog signals. The HSA seems to work well when the values of the waveforms are large and do not take values close to zero, and do not change too quickly relative to the time width of the convolution filter window. An example of applying too stringent an analog wave form is shown in Fig.3. It may be possible to simply add a constant value to incoming analog signals before spiking them and to ensure that the analog signal does not change too rapidly.

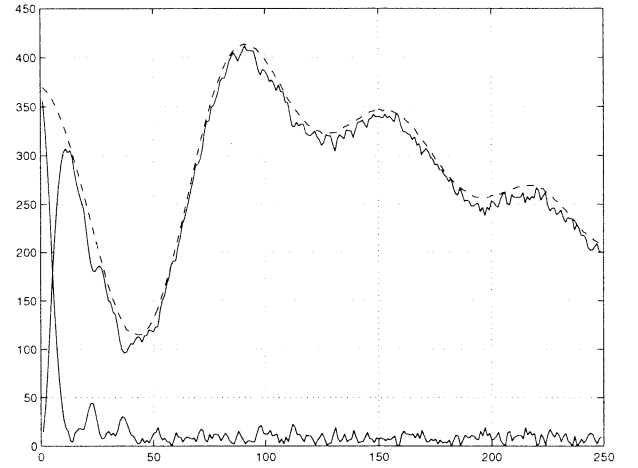


Figure 2: The original curve (dashed line) is converted into a spiketrain by the HSA. The resulting spiketrain is converted into an analog waveform, using the SIIC (solid line). The lower curve shows the absolute error between the two.

## 4 Summary and Conclusions

This paper introduced an algorithm, the Hough Spiker Algorithm (HSA), to transform analog waveforms into digital spiketrains (bitstrings) which will be very useful to designers of the component modules of the artificial brains containing tens of thousands of such modules. Used in conjunction with the Spike Interval Information Coding (SIIC), it is possible to think entirely in terms of analog waveforms, which is considerably easier than dealing with abstract spiketrains. After an input waveform is spec-

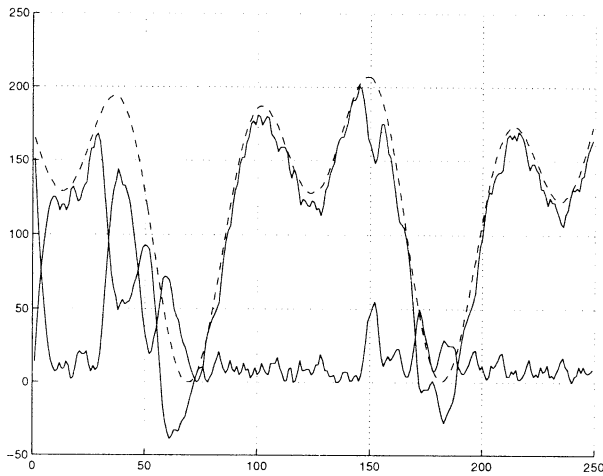


Figure 3: Another example of original waveform (dashed line)  $\xrightarrow{HSA}$  spiketrain  $\xrightarrow{SIIC}$  estimated waveform (solid line)

ified, it is then converted into spiketrains, using the HSA. These spiketrains are fed into the CoDi modules being evolved. Now, the spiketrains output by the CoDi modules can then be converted into an analog waveform using the reverse process of the HSA, the SIIC. The resulting waveform output can then be compared with a target analog waveform. The closer the match, the higher the fitness of the evolving module. These two conversions, analog to spiketrain (HSA) and spiketrain to analog (SIIC) facilitate the evolution of CoDi neural net modules on the CAM-Brain Machine (CBM).

With these two conversion algorithms on hand, it will be a lot more practical for designers to evolve CoDi modules, provided of course that the evolvability of the modules is adequate. The issue of evolvability is always an open question, because not all modules may evolve equally well. Criteria for good evolvability, or how to enhance it, are still not well understood issues.

The next step in the CAM-Brain Project is to design multi-module systems, and to scale up the number of modules used. The CBM can update roughly 32000 modules at sufficient speed (150 billion cellular automata cells a second) to enable real time control of a kitten robot. With 32000 modules allowed by the hardware, more sophisticated neural network modules, with more interesting behaviors, and multi-module architectures can be tried. The CBM will allow the systematic testing of module evolution procedures, which is rather difficult to be done on standard computers due to the long simulations times.

## References

- [1] Hugo de Garis. An artificial brain : ATR's cam-brain project aims to build/evolve an artificial brain with a million neural net modules inside a trillion cell cellular automata machine. *New Generation Computing Journal*, 12(2), July 1994.
- [2] Felix Gers and Hugo de Garis. CAM-Brain: A new model for ATR's cellular automata based artificial brain project. In *Proceedings of the First International Conference on Evolvable Systems: From Biology to Hardware (ICES'96)*, October 1996.
- [3] Michael Korkin, Hugo de Garis, Felix Gers, and Hitoshi Hemmi. CBM (CAM-Brain Machine): A hardware tool which evolves a neural net module in a fraction of a second and runs a million neuron artificial brain in real time. In John R. Koza, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max Garzon, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1997: Proceedings of the Second Annual Conference*, July 1997.
- [4] Michael Korkin, Norberto Eiji Nawa, and Hugo de Garis. A 'spike interval information coding' representation for atr's CAM-brain machine (cbm). In *Proceedings of the Second International Conference on Evolvable Systems: From Biology to Hardware (ICES'98)*. Springer-Verlag, September 1998.
- [5] Norberto Eiji Nawa, Hugo de Garis, Felix Gers, and Michael Korkin. ATR's CAM-brain machine (CBM) simulation results and representation issues. In John R. Koza, Wolfgang Banzhaf, Kumar Chellapilla, Kalyanmoy Deb, Marco Dorigo, David B. Fogel, Max H. Garzon, David E. Goldberg, Hitoshi Iba, and Rick L. Riolo, editors, *Genetic Programming 1998: Proceedings of the Third Annual Conference*. San Francisco, CA, Morgan Kaufmann, July 1998.
- [6] Fred Rieke, David Warland, Rob de Ruyter van Steveninck, and William Bialek. *Spikes: exploring the neural code*. MIT Press/Bradford Books, Cambridge, MA, 1997.

## Generation of Jumping Motion Pattern for Hopping Robot using Genetic Algorithm

Y. Yoshida, T. Kamano, T. Yasuno, and T. Suzuki

Dept. of Electrical and Electronic Eng.

The University of Tokushima

2-1 Minamijyosanjinima-cho, Tokushima-city, 770-8506, Japan

Y. Kataoka

Kataoka Machine Co., LTD.

4765-46 Sangawa-cho, Iyomishima-city, 799-0431, Japan

**Key words:** Genetic algorithm, Hopping robot, Adaptation, Pattern generator

### Abstract

In this paper, an application of genetic algorithm for generation of a jumping motion pattern of a hopping robot is described. The tuning parameters of a central pattern generator are regarded as genes and adjusted by the genetic algorithm so that the hopping robot can continuously jump to the reference height with the minimum force. To realize online tuning of the parameters, new genetic operations are introduced. Experimental results demonstrate the effectiveness of the proposed scheme for generation of the suitable jumping motion pattern.

### 1 Introduction

Physiological experiments suggest that periodic motion patterns such as walking and flapping are generated by Central Pattern Generators (CPGs) <sup>1</sup>. To generate the most suitable motion patterns, one required function of the CPG is adaptation to the growth of the muscles and the frameworks. Some adaptation schemes of the CPG have been reported <sup>2, 3, 4, 5</sup>. In this paper, an application of genetic algorithm to realize adaptation between the CPG and a hopping robot is considered.

Generally, the genetic algorithm is a search and/or an optimization algorithm based on the natural evolution mechanism, and has been successfully applied to several optimization problems <sup>6</sup>.

The objective of the genetic algorithm in this paper is to tune the CPG so that the most suitable motion pattern for the hopping robot is generated. If the successful adaptation is achieved, the most suitable pattern is obtained and the hopping robot can continuously jump to the reference position with the minimum force.

In the proposed system, the CPG acts as a feedback controller and of which parameters are regarded as genes. To achieve online tuning of the genes, new genetic operations such as a few individuals, quick es-

timation, instant selection and intentional mutation is introduced. The effectiveness of the proposed genetic algorithm for successful adaptation between the CPG and the hopping robot is demonstrated by experiment.

### 2 System Configuration

#### 2.1 Hopping robot

Fig.1 shows the experimental setup. The hopping robot consists of a dc servo motor (35[W], 24[V], 2.5[A], 3500[rpm]) mounted on a platform, a crank and a spring attached to the platform.

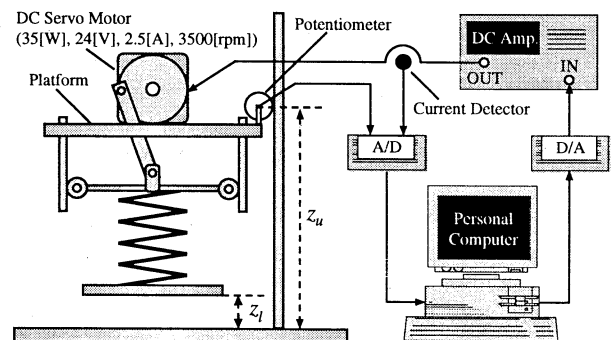


Fig. 1: Hopping robot.

The motor is supplied by a current amplifier, of which output signal corresponds to the reference torque. The motor torque is converted into the periodic force to the spring. The position of the platform,  $z_u$ , is measured by a potentiometer. The measured position is used to calculate the velocity,  $v_u$ , and the acceleration,  $a_u$ , of the platform. The motor current is also measured to evaluate the motor torque. When the suitable force is applied to the spring, the platform can make a continuous jumping by the use of the repulsive force.

## 2.2 Central pattern generator

The overall control system under the proposed scheme is shown Fig.2. The objective of the control system is to generate the adequate motion pattern, which makes the continuous jumping to the desired height with the minimum force. Therefore, the adjustment of the applied force in synchronization with the robot motion is required. In this system, the CPG acts as a feedback controller, and the position,  $z_u$ , the velocity,  $v_u$ , and the acceleration,  $a_u$  are used as the input signals of the CPG to realize the synchronization between the CPG and the robot. By defining,

$$\phi^T = [z_u, v_u, a_u] \quad (1)$$

and

$$\mathbf{W}^T = [W_0, W_1, W_2] \quad (2)$$

the torque reference generated by the CPG is given by

$$i_r = \phi^T \mathbf{W} \quad (3)$$

where  $\phi$  is the input vector and  $\mathbf{W}$  is the parameter vector.  $\mathbf{W}$  is adjusted by the genetic algorithm so that both the position error in each jumping and the motor current are minimized.

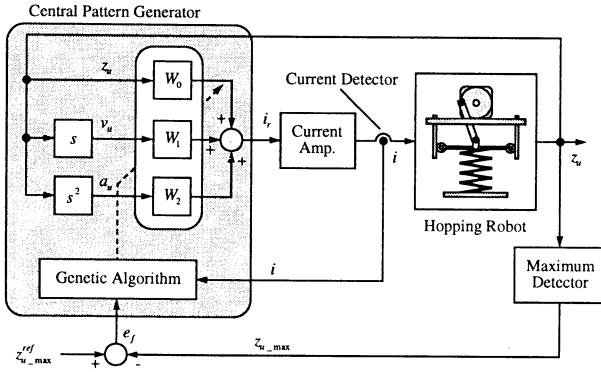


Fig. 2: System configuration.

## 3 New Genetic Algorithm

### 3.1 Encoding

The parameter vector  $\mathbf{W}$  corresponds to the genes, and represents the organization of an individual,  $n$ , of a group of the population,  $N$ . Then, the individual,  $n$ , is given by the real number genetic codes as

$$n: W_0 \ W_1 \ W_2$$

By applying to the CPG in order, the individual is estimated at every jumping motion and adjusted by the following selection, crossover and mutation operations.

### 3.2 Estimation

To estimate the individuals, the position error is defined by

$$e_f = z_{u\_max}^{ref} - z_{u\_max} \quad (4)$$

where  $z_{u\_max}$  is the highest position in each jumping and the  $z_{u\_max}^{ref}$  is the reference position. Each individual is evaluated by using the position error,  $e_f$ , and the integrated value of the motor current,  $i$ . Then, the fitness of the individual,  $n$ , is given by

$$f_n(k) = |e_f(k)| + g \int_0^{T_s} |i(t_k + t)| dt \quad (5)$$

where  $k$  is the number of jumping, and  $T_s$  is the jumping period and  $t_k$  is the beginning of the  $k$ -th jumping. If the weight parameter,  $g$ , is set to zero, the individuals are estimated by only position error,  $e_f$ . Eqn.(5) demonstrates that the fitness  $f_n$  is positive value, and the smaller fitness gives the better estimation of the individual.

### 3.3 Genetic operations

Fig.3 shows the outline of the proposed genetic algorithm. To realize quick alternation of generations, the number of individuals in each generation is set to four.

Firstly, the mutant denoted by  $[a', b', c', d']$  of four individuals  $[a, b, c, d]$  are obtained by the intentional mutation. These individuals are applied and evaluated by Eqn.(5) in order of  $a, a', b, b', c, c', d, d'$ . As a result, if  $f_a > f_{a'}$ ,  $f_b > f_{b'}$ ,  $f_c < f_{c'}$  and  $f_d > f_{d'}$ , the individuals  $[a, b, c', d]$  are not able to remain in the next generation. This operation prevents losing the better individuals by inadequate mutation.

Next, the remaining individuals  $[a', b', c, d']$  are ranked by the fitness. As a result of ranking, if  $f_c < f_{a'} < f_{d'} < f_b$ , the best individual  $[c]$  remains in the next generation without modification. The rest of the individuals in the next generation are randomly created by combining the best individual  $[c]$  and two individuals in  $[a', d, b']$  based on ranking strategy. This crossover operation results in high speed convergence of the tuning parameters because the best individual rapidly spreads in the following generation. Consequently, new four individuals  $[c, ca', ca'd', cd']$  are obtained by the genetic operation.

### 3.4 Intentional mutation

To ensure the convergence of the tuning parameters, the intentional mutation is introduced. Then,



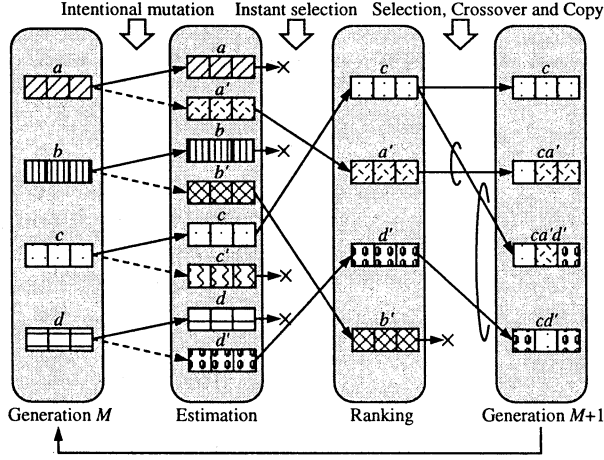


Fig. 3: Genetic operation

the mutant  $[a', b', c', d']$  are obtained by multiplying the following coefficient,  $d_m$ , to  $[a, b, c, d]$ ,

$$d_m = p e_f + q \quad (6)$$

where  $p$  and  $q$  are constants. Eqn.(6) implies that the mutation range is depend on the position error and becomes narrow as the position error decreases.

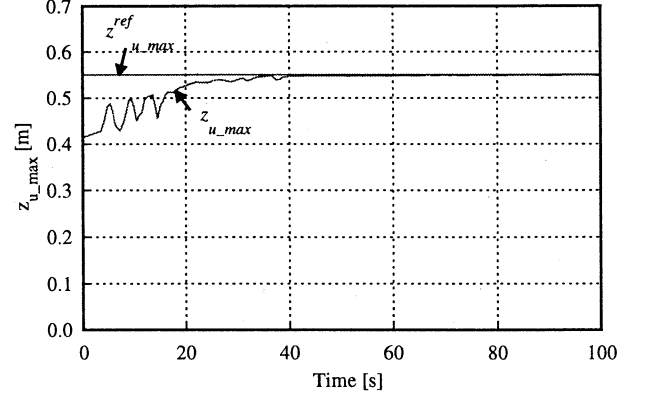
#### 4 Experimental results

To confirm the validity of the proposed scheme, the responses of the hopping robot in both cases of with and without current estimation. When the current estimation is added,  $g$  is set to  $2.50 \times 10^{-3}$ .

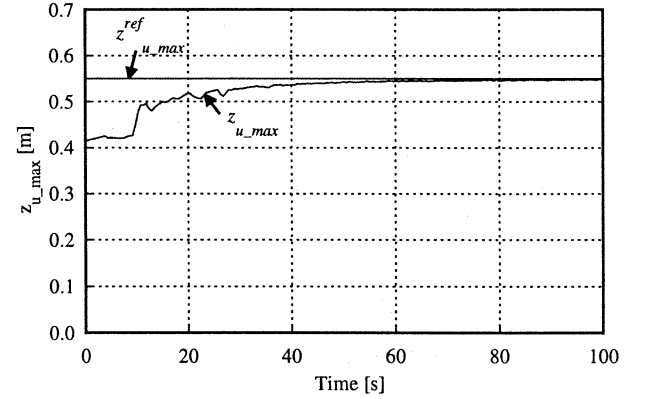
Figs.4(a) and 4(b) show the loci of the highest position in each jumping for the same constant reference position 0.55[m]. After the tuning process is completed, the highest positions in both cases converge to the reference position. This implies that the hopping robot can continuously jump to the reference position after tuning.

Fig.5 shows the responses of the position and the waveforms of the motor current after tuning process. Although the robot can jump in the same manner, the motor current in case of with current estimation is less than that in case of without current estimation. These results show that the suitable jumping motion pattern is generated by adding the current estimation.

Figs.6(a), 6(b) and 6(c) show the responses of the tuning parameters. In early stage of tuning, the parameters fluctuate in an irregular manner due to genetic operations. However, the parameters converge to the constant values after about 80[s].



(a) with current estimation



(b) without current estimation

Fig. 4: loci of the highest position

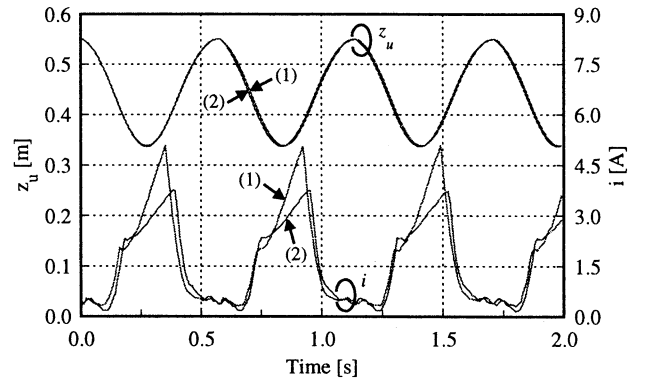


Fig. 5: Responses of the position and the motor current.

- (1) without current estimation
- (2) with current estimation

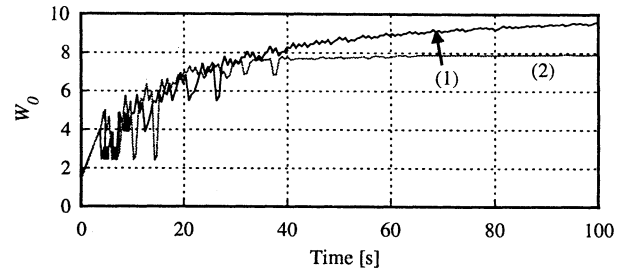
Fig.7 shows the responses of the fitness defined by Eqn.(5). The fitness decreases to the constant value which is the integrated value of the motor current. This result also demonstrates the effectiveness of the proposed scheme for generation of the jumping motion pattern.

## 5 Conclusions

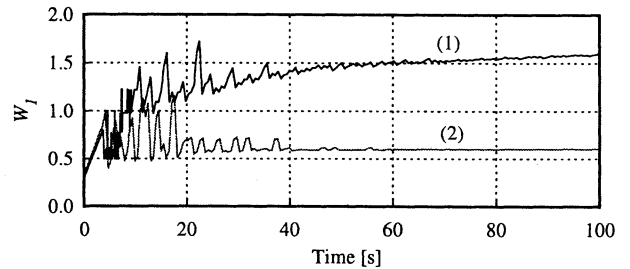
In this paper, the use of genetic algorithm for generation of jumping motion pattern of a hopping robot has been considered experimentally. The genetic algorithm was used to adjust the tuning parameters of the CPG so that the hopping robot could jump to the reference position with the minimum driving force. To realize on-line tuning, new genetic operations such as a few individuals, rapid estimation, instant selection and intentional mutation were introduced. Experimental results demonstrated that the proposed scheme was effective for generation of the most suitable motion pattern.

## References

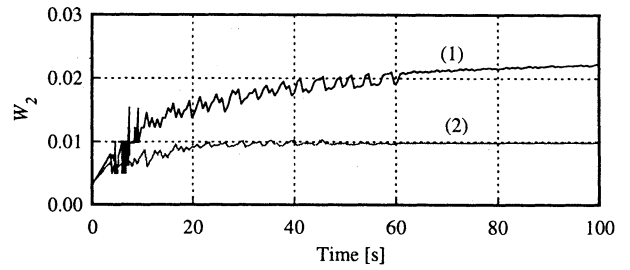
- [1] Sten Grillner, Neurobiological Bases Rhythmic Motor Acts in Vertebrates, *Science*, 228, pp.143-149 (1985)
- [2] A.H.Cohen, P.J.Holmes, R.H.Rand, The Nature of the Coupling Between Segmental Oscillators of the Lamprey Spinal Generator for Locomotion : A mathematical Model, *Journal of Mathematical Biology*, 13, pp.345-369 (1982)
- [3] J.J.Colins, S.A.Richmond, Hard-wired central pattern generators for quadrupedal locomotion, *Biological Cybernetics*, 71, pp.375-385 (1994)
- [4] S.Kimura, M.Yano, H.Shimizu, A self-organizing model of walking patterns of insects, *Biological Cybernetics*, 69, pp.183-193 (1993)
- [5] S.Kimura, M.Yano, H.Shimizu, A self-organizing model of walking patterns of insects II, *Biological Cybernetics*, 70, pp.505-512 (1994)
- [6] Hiroaki Kitano, Genetic Algorithm, *Sangyotosyo*, in Japanese, (1993)



(a)  $W_0$



(b)  $W_1$



(c)  $W_2$

Fig. 6: Responses of the tuning parameters.  
(1) without current estimation  
(2) with current estimation

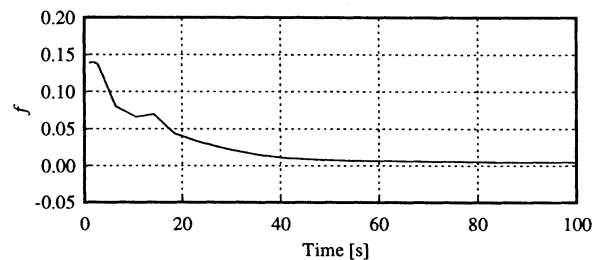


Fig. 7: Fitness

# An Evolutionary Optimal Obstacle Avoidance Method for Mobile Robots

M.M.A. Hashem\*, Keigo Watanabe\*\* and Kiyotaka Izumi\*\*\*

\*Faculty of Engineering Systems and Technology,  
Graduate School of Science and Engineering,

\*\*Department of Advanced Systems Control Engineering,  
Graduate School of Science and Engineering,

\*\*\*Department of Mechanical Engineering,  
Faculty of Science and Engineering,  
Saga University, 1-Honjomachi, Saga 840-8502, Japan

## Abstract

In this paper, an evolutionary optimal obstacle avoidance method based on a Novel Evolution Strategy (NES) algorithm is presented. The obstacle avoidance problem is formulated as a constrained time-optimal control problem considering the motions of a point robot. The polygonal obstacles are represented as circles from the visibility and sensor modeling concepts. Some special representations and operators are adopted for the success of the evolutionary search. The effectiveness of this method is illustrated by some simulations of the robot within a heavily obstacle environment.

**Key Words:** Evolutionary Algorithms; Path Planning; Optimal Control; Constrained Optimization; Intelligent Robotic Systems.

## 1 Introduction

Finding the shortest path from a given position to the goal position among stationary polygonal obstacles is a classical problem for intelligent robotic navigation. The efficiency of a path planning algorithm largely depends upon the representation of obstacles encountered by the robot as well as upon the path-searching strategy. Mobile robot path planning problems can generally be classified as local or global, depending on the environment currently surrounding the robot. Several methods exist in the literature [1]~[3] for this purpose. Among these methods, optimal control methods [2, 3] are attractive due to their robustness and they can directly yield the optimal controls required for a robot to move on the path while avoiding the obstacles.

In this paper, the obstacle avoidance problem among polygonal obstacles is formulated as a constrained time-optimal control problem considering the motions of the robot. An evolutionary algorithm based on a Novel Evolution Strategy (NES) [5] algorithm is developed to solve this problem. The polygonal obstacles

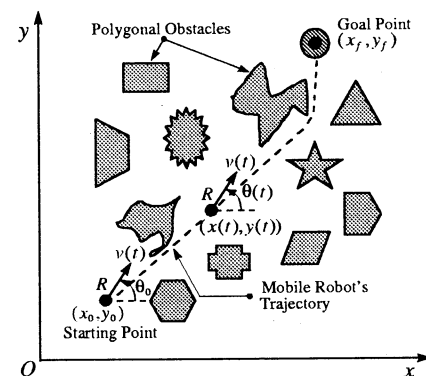


Figure 1: A two-dimensional world model where a point mobile robot is traveling among polygonal obstacles

in the environment are represented as circles from the "visibility" [3] and sensor modeling [4] concepts. In contrast to the traditional representations, some special representations and operators are adopted for the success of the evolutionary search. The effectiveness of this method is illustrated by some simulations of the robot within a heavily obstacle environment.

## 2 Problem Formulation

Consider an optimal obstacle avoidance problem as shown in **Fig. 1**, where the current position and velocity of a point mobile robot  $\mathcal{R}$  are represented by the Cartesian coordinates  $(x(t), y(t))$  and  $v(t)$  on the world coordinate frame  $O - xy$ , where  $t$  is time. The starting and goal points of the robot are  $(x_0, y_0)$  and  $(x_f, y_f)$ , and its steering angle is  $\theta(t) \in [-\pi, \pi]$ , which is measured from the  $x$ -axis. The equations of motion for the robot  $\mathcal{R}$  can be expressed as

$$\dot{x}(t) = v(t) \cos \theta(t) \quad (1)$$

$$\dot{y}(t) = v(t) \sin \theta(t) \quad (2)$$

There are polygonal obstacles in the plane of its motion. The objective is to plan an optimal path from

start to goal while avoiding the obstacles with minimum control effort and time/distance. With these constraints, the path planning problem can be formulated as a time-optimal control problem:

$$\min_{\theta(t)} \int_0^{t_f} 1 dt \quad \text{where } t_f \text{ is not specified} \quad (3)$$

subject to the system dynamics (1)-(2) with the velocity constraint such that  $v(t)$  is essentially a constant  $V$ , the obstacle constraints:

$$g(x(t), y(t)) \geq 0; \quad g \in R^q \quad (4)$$

where  $q$  is the number of obstacles, and the boundary conditions:

$$(x(0), y(0)) = (x_0, y_0); \quad (5)$$

$$(x(t_f), y(t_f)) = (x_f, y_f) \quad (6)$$

The criterion (3) refers to a unique shortest path that ensures the control effort and time/distance to be minimized. In the discrete-time, the motion equations (1)-(2) can be approximated as

$$x(k+1) = x(k) + V\Delta T \cos \theta(k) \quad (7)$$

$$y(k+1) = y(k) + V\Delta T \sin \theta(k) \quad (8)$$

for  $k = 0, 1, \dots, (N-1)$ , where  $\Delta T$  is the sampling width,  $k$  is a sampling instant and  $N$  is the terminal sampling instant.

Thus, the problem of (3) is reduced to

$$\min_{\theta(k)} \sum_{k=0}^{N-1} k \quad \text{where } N \text{ is not specified} \quad (9)$$

subject to the discrete system dynamics (7)-(8), the obstacle constraints:

$$g(x(k), y(k)) \geq 0; \quad g \in R^q \quad (10)$$

and the boundary conditions:

$$(x(0), y(0)) = (x_0, y_0); \quad (11)$$

$$(x(N), y(N)) = (x_f, y_f) \quad (12)$$

### 3 Special Considerations

#### 3.1 Individual Representation

Since the  $N$  is not specified, then different time-length individuals within some bound are needed to evolve. Therefore, an individual  $\psi$  is represented by  $\theta$  values for all sampling instants as

$$\psi = [\theta(0), \theta(1), \dots, \theta(N_\psi - 1)] \quad (13)$$

where  $N_\psi$  is the terminal sampling instant for an individual  $\psi$  generated from a given bound.

#### 3.2 SBMAC Representation

The SBMAC (Subpopulation-Based Max-mean Arithmetical Crossover) operator employed in the NES algorithm [5] can be used only for equal length individuals. However, the formulated problem consists

of different length individuals, so an operational modification is needed to fit this concept for the problem. Firstly, the minimum-length individual within each subpopulation is determined. Secondly, the offspring variables are produced up to this length. Then, the remaining part of the parent variables is appended to the offspring variables to form the full length chromosome.

#### 3.3 Obstacle Representation

From the “visibility” [3] and sensor modeling [4] concepts, each polygonal obstacle in the environment is represented by a circle with a center of  $(x^c, y^c)$  and a radius of  $r$  so that the visible side of the obstacle will be contained inside a circle.

#### 3.4 Variation Operators

##### 3.4.1 Swapping Crossover

The swapping crossover is also proposed to produce hopefully obstacle free chromosomes. Two offspring chromosomes, selected with a probability of crossover  $P_c$ , undergo this operation. If two selected offspring are different in length, then a crossover point is randomly selected from the minimum-length individual, otherwise it is selected from one of their lengths. Thus two selected offspring are cut at this crossover point and swapped their gene values to form the full length chromosome.

##### 3.4.2 Insertion and Deletion Mutations

These type of mutations are incorporated to vary the length of individuals as well as hopefully to produce feasible solutions. For insertion mutation, a new randomly generated gene is inserted at a position selected by a probability of insertion  $P_{im}$ . This will increase the length of a chromosome and hopefully will produce a feasible solution. For deletion mutation, an existing gene is deleted from a position selected by a probability of deletion  $P_{dm}$ . This mutation will also produce hopefully a feasible solution and will decrease the length of that chromosome.

### 4 Fitness Function Construction

The fitness function is the only link between the evolutionary algorithm and environment. The penalty function [6] based fitness function for an individual  $\psi$  of length  $N_\psi$  is constructed as

$$\Phi(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}, \mathbf{y}) + \frac{s_t}{2} \left[ \left\{ \sum_{k=1}^{N_\psi} (g_1^+(k))^2 + \dots, \right. \right. \\ \left. \left. + \sum_{k=1}^{N_\psi} (g_q^+(k))^2 \right\} + \sum_{j=1}^2 (h_j)^2 \right] \quad (14)$$

where  $f(\mathbf{x}, \mathbf{y})$  is the total distance produced by the individual and defined by

$$f(\mathbf{x}, \mathbf{y}) = \sum_{k=0}^{N_\psi-1} \sqrt{\Delta x^2(k) + \Delta y^2(k)}$$

where  $\Delta x(k) = x(k+1) - x(k)$ ,  $\Delta y(k) = y(k+1) - y(k)$  with position coordinates of the robot  $(x(1), y(1)), \dots, (x(N_\psi), y(N_\psi))$  calculated using (7)-(8) by the individual's chromosome  $[\theta(0), \theta(1), \dots, \theta(N_\psi - 1)]$ . Here,

$g_1^+(x(k), y(k)), \dots, g_q^+(x(k), y(k))$  are the magnitudes of obstacle constraint (modeled as circles and centered at  $(x_1^c, y_1^c), \dots, (x_q^c, y_q^c)$  with radius  $r_1, \dots, r_q$  respectively) violations by the individual and evaluated as  $g_1^+(k) = \max\{0, (-(x(k) - x_1^c)^2 - (y(k) - y_1^c)^2 + r_1^2)\}, \dots, g_q^+(k) = \max\{0, (-(x(k) - x_q^c)^2 - (y(k) - y_q^c)^2 + r_q^2)\}$  where  $k = 1, \dots, N_\psi$ ;  $h_j$ 's are the end point equality constraint violation measures by the individual defined as  $h_1 = x_f - x(N_\psi)$ ,  $h_2 = y_f - y(N_\psi)$ ; and  $s_t$  is an iteration dependent penalty parameter defined as  $s_t = s_0 + \log(t + 1)$  where  $s_0$  is an initial penalty parameter value and  $t$  is the current generation number.

## 5 Parameter Bounds

### 5.1 N Bounds

According to the problem, it is necessary that any obstacle free optimal path from  $(x_0, y_0)$  to  $(x_f, y_f)$  should abide by following:

$$N_\psi^* = \frac{\sum_{k=0}^{N_\psi^*-1} \sqrt{\Delta x^2(k) + \Delta y^2(k)}}{V \Delta T} \quad (15)$$

where  $N_\psi^*$  is the obtained optimal terminal sampling instant for an individual  $\psi$ .

By (15), the  $N$  bounds  $[N_l, N_u]$  can easily be determined. The nominal distance from start to goal point defined by  $\sqrt{(x_f - x_0)^2 + (y_f - y_0)^2}$  can be used for  $N_l$ . And for  $N_u$ , as an example, the goal coordinate summing distance  $(x_f + y_f)$  can be used. Thus the lower and upper bounds of  $N$  can be determined by

$$N_l = \left\lfloor \frac{\sqrt{(x_f - x_0)^2 + (y_f - y_0)^2}}{V \Delta T} \right\rfloor \quad (16)$$

$$N_u = \left\lceil \frac{x_f + y_f}{V \Delta T} \right\rceil \quad (17)$$

where  $\lfloor \cdot \rfloor$  and  $\lceil \cdot \rceil$  denotes the floor and ceiling functions respectively.

As that of genetic algorithms [6], the probabilities of insertion and deletion mutations can be calculated from the lower bound  $N_l$  as

$$P_{im} = P_{dm} = \frac{1}{N_l} \quad (18)$$

### 5.2 $\theta$ Bounds

In general, the  $\theta$  bounds  $[\theta_l, \theta_u]$  are  $[-\pi, \pi]$ . But they can be set up to their optimum values for a particular starting and goal positions with the help of obstacle distributions within the environment. The optimum  $\theta$  bounds are given by

$$\theta_u = \beta + \alpha_1 \quad (19)$$

$$\theta_l = \beta - \alpha_2 \quad (20)$$

where  $\beta$  is an initial positional angle of the robot with respect to the goal measured from the  $x$ -axis of the world coordinate frame as

$$\beta = \tan^{-1} \frac{(y_f - y_0)}{(x_f - x_0)} \quad (21)$$

Here,  $\alpha_1$  is the right-ward obstacle tangential angle with  $\beta$  for which the tangent passes through and  $\alpha_2$  is the left-ward obstacle tangential angle with  $\beta$  for which the tangent passes through.

## 6 Proposed Evolutionary Algorithm

The evolutionary obstacle avoidance algorithm based on the NES algorithm [5] is given here. The parameters of the algorithm like  $\mu$ ,  $T$ ,  $\gamma$ ,  $l$ ,  $P_c$  and  $s_0$  will be supplied in advance. The procedure of the algorithm goes as follows:

1. Calculate the  $N$  bounds  $[N_l, N_u]$  using (16)-(17) and the  $\theta$  bounds  $[\theta_l, \theta_u]$  using (19)-(20), and the probabilities  $P_{im}$  and  $P_{dm}$  using (18).
2. Set the generation counter  $t = 0$ .
3. Create the initial population consisting of  $\mu$  individuals by the following procedure:
  - (a) Determine each individual's length  $N_\psi$  from the  $\text{URN}[N_l, N_u]$ .
  - (b) Create each individual chromosome  $\psi = [\theta(0), \theta(1), \dots, \theta(N_\psi - 1)]$  from the  $\text{URN}[\theta_l, \theta_u]$  up to its length  $N_\psi$  found in previous step.
4. Evaluate the initial population by the fitness function  $\Phi(\mathbf{x}, \mathbf{y})$  and order this population according to their fitness values.
5. Apply the modified SBMAC on the parent population to produce  $\mu$  offspring and apply the TVM operator on each offspring.
6. Apply the swapping crossover on this offspring population with a crossover probability,  $P_c$ .
7. Apply the following special mutation operators on the offspring simultaneously:
  - (a) Apply the insertion mutation operator with a probability,  $P_{im}$ .
  - (b) Apply the deletion mutation operator with a probability,  $P_{dm}$ .
8. Evaluate the offspring population with the fitness function  $\Phi(\mathbf{x}, \mathbf{y})$ .
9. Alternate the generation by ordering the combined offspring and parent populations, and then by selecting the best  $\mu$  individuals as parents for the next generation.
10. Increase the generation counter by one i.e.,  $t = t + 1$ .
11. If  $f(\mathbf{x}, \mathbf{y}) \neq \Phi(\mathbf{x}, \mathbf{y})$  for the best individual or  $t \neq T$ , then go to step 5, else stop and plan the trajectory with the best individual's chromosome using (7) and (8).

## 7 Simulations and Results

The proposed algorithm was implemented using  $\mu = 100$ ,  $l = 10$ ,  $\gamma = 8.0$ ,  $s_0 = 1 \times 10^3$ ,  $P_c = 0.6$ ,  $T = 1000$ ,  $V = 0.25$  [m/s] and  $\Delta T = 0.1$  [sec]. A simulation world frame with 7 polygonal obstacles modeled as circles was defined by  $1\text{m} \times 1\text{m}$ . Modeled obstacle center positions and radii are given in **Table 1**.

Table 1: Modeled obstacle sizes and positions

Obstacle No.	Radius [m]	Position ( $x^c, y^c$ )
1	0.140	0.40, 0.20
2	0.120	0.55, 0.55
3	0.125	0.80, 0.15
4	0.170	0.20, 0.50
5	0.150	0.80, 0.80
6	0.125	0.45, 0.85
7	0.130	0.85, 0.45

To demonstrate the effectiveness of the proposed method for optimal obstacle avoidance, two simulations were conducted such as local and global maneuvers. For the local one, only the obstacles in the vicinity of the subgoals *A*, *B* and *C* (**Fig. 2**) were taken into account while for the global one, all the obstacles in the environment were considered at a time. The simulation parameters for the local maneuver is shown in **Tables 2**. Without loss of generality, starting and goal positions for the global one were assumed as (0, 0) and (1, 1) respectively. For this maneuver, the bounds for  $[\theta_l, \theta_u]$  and  $[N_l, N_u]$  were calculated as  $[0, \frac{\pi}{2}]$  and  $[56, 80]$  respectively. Consequently, the insertion and deletion mutation probabilities  $P_{im}$  and  $P_{dm}$  became as 0.0179.

Table 2: Local maneuver parameters for the algorithm

Start	Goal	$[\theta_l, \theta_u]$	$[N_l, N_u]$	$P_{im}/P_{dm}$
<i>S</i> (0.00, 0.00)	<i>A</i> (0.60, 0.30)	$[0, \frac{\pi}{2}]$	[18, 36]	0.0555
<i>A</i> (0.60, 0.30)	<i>B</i> (1.00, 0.40)	$[0, \frac{\pi}{2}]$	[16, 56]	0.0625
<i>B</i> (1.00, 0.40)	<i>C</i> (0.65, 0.65)	$[-\pi, \pi]$	[17, 52]	0.0588
<i>C</i> (0.65, 0.65)	<i>G</i> (1.00, 1.00)	$[0, \frac{\pi}{2}]$	[19, 80]	0.0526

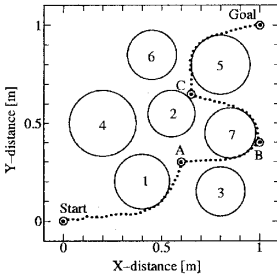


Figure 2: The optimal obstacle avoidance by the robot for the local maneuver

The simulation results for both the maneuvers are shown in **Figs. 2~3**. The optimal steps  $N$  found for the local maneuver were 30, 17, 19 and 24, and that for the global one was 60. These results indicate that

the paths are completely obstacle free and they are optimal. Thus the mobile robot was guaranteed to reach the respective destination with minimum time.

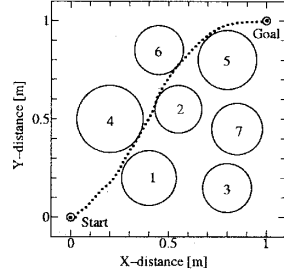


Figure 3: The optimal obstacle avoidance by the robot for the global maneuver

## 8 Concluding Remarks

In this paper, an evolutionary technique for optimal obstacle avoidance has been presented for a point mobile robot considering its motions. The problem was composed of mixed integer and discrete constrained optimization problem. It was really difficult to solve such a problem with the conventional calculus-based methods. The method responded well for all the simulation attempts. The evolutionary method seemed to be robust in the sense that it was guaranteed to yield a trajectory terminating at the goal with minimum time and distance while avoiding obstacles.

## References

- [1] Al-Sultan, K.S. and Aliyo, M.D.S., "A New Potential Field-Based Algorithm for Path Planning," Journal of Intelligent and Robotics Systems, Vol. 17, pp. 265-282, (1996).
- [2] Sundar, S. and Shiller, Z., "Optimal Obstacle Avoidance Based on the Hamilton-Jacobi-Bellman Equation," IEEE Trans. on Robotics and Automation, Vol. 13, No. 2, pp. 305-310, (1997).
- [3] Tsoularis, A. and Kambhampati, C., "On-line Planning for Collision Avoidance on the Nominal Path," Journal of Intelligent and Robotics Systems, Vol. 21, pp. 327-371, (1998).
- [4] Fort-Piat, N.L., Collin, I. and Meizel, D., "Planning Robust Displacement Missions by Means of Robot-Task and Local Maps," Robotics and Autonomous Systems, Vol. 20, pp. 99-114, (1997).
- [5] Hashem, M.M.A., Watanabe, K. and Izumi, K., "A New Evolution Strategy and Its Application to Solving Optimal Control Problems," JSME International Journal, Series C, Vol. 41, No. 3, pp. 406-412, (1998).
- [6] Michalewicz, Z., "Genetic Algorithms + Data Structures = Evolution Programs," 3rd. rev., and extended edition, Springer-Verlag, Berlin, (1996).

# Design and Experiment of an Omnidirectional Mobile Robot Using Fuzzy Servo Control

Jun Tang<sup>‡</sup>, Keigo Watanabe<sup>‡</sup> and Katsuhori Sanefuji<sup>‡</sup>

<sup>‡</sup> Department of Mechanical Engineering, Yamaguchi University,  
Tokiwadai-2557, Ube 755-8611, Japan

<sup>‡</sup> Department of Advanced Systems Control Engineering, Saga University,  
Honjomachi-1, Saga 840-8502, Japan

## Abstract

The paper presents a new approach how to achieving full mobility in the design of omnidirectional mobile robot, which is to use the single spherical-wheel assembly. Here, a ball wheel mechanism that allows for full mobility and smooth motion is developed. We also proposed a fuzzy linear models control method to solve this robot's nonlinear control problem. Finally, we present the results of experiment and show how to use our method to control the omnidirectional mobile robot.

## 1. Introduction

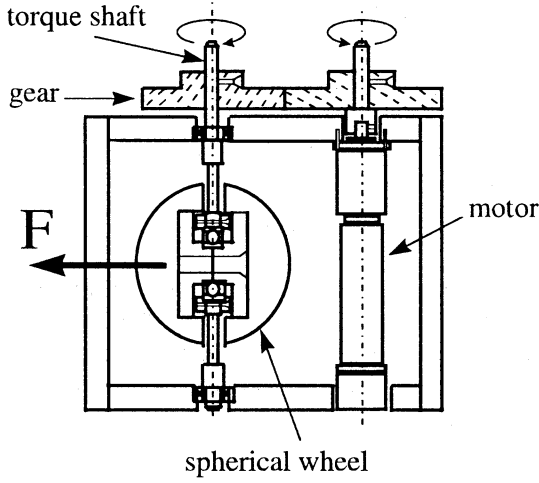
Automated guided vehicles (AGV) and wheels mobile robot (WMR) have been used in automated factories for material handling. If they can also be used in other narrow areas such as warehouse and hospital services, the robot will be required to possess special mechanical structure with a full mobility. Therefore, the studies for the omnidirectional mobile robot have an attractive feature, and some kinds of mechanisms for omnidirectional holonomic mobile robot have been already considered: the driving wheel with steering (or offset driving wheel); universal wheels; spherical tires; crawler mechanisms; and orthogonal-wheel assembly [1].

The paper presents an another approach how to achieving full mobility in the design of omnidirectional mobile robot, which is to use the single spherical-wheel assembly. For the mobile robot, the state variables assigned to the dynamic model are composed of two-dimensional positions, velocities, and a rotational angle between the moving coordinates and the absolute coordinates, and its rate of angle. One of the features

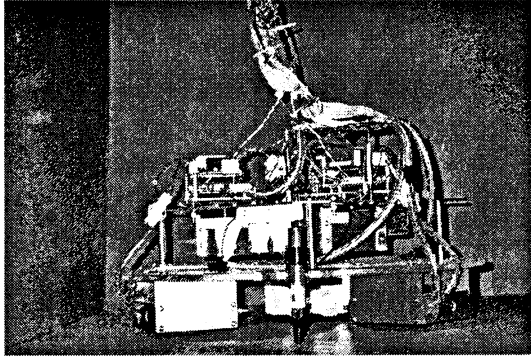
to this model is that it is nonlinear with respect to the rotational angle. We also present a fuzzy linear models control method to solve the nonlinear control problem [2]. In order to build the model-based fuzzy servo system, we first choose some representative rotational angles to simplify the nonlinear model to multiple linear models. Then, optimal type-1 servo controllers are derived for the rotational control of the mobile robot. Finally, we present the results of experiment and show how to use our method to control the omnidirectional mobile robot.

## 2. Omnidirectional mobile robot and its Dynamics

As shown in Fig. 1, a ball wheel mechanism that allows for full mobility and smooth motion is developed. The ball is cut into halves, then two vertical axles from each other are mounted at the centre of the ball. The two half balls can freely rotate around the axle which links up the two sliced surfaces. The other axle installed in perpendicular position is driven by DC servo motor. Three same assemblies are respectively allocated at the each apex of a regular triangle: the angle offset between two assemblies is 120°. Figure 2 shows the mobile robot, whose platform consists of a disc with 500 [mm] diameter. The bracket of assembly is 65 [mm] in height, 114 [mm] in width and 105 [mm] in length, and the sphere wheel has 60 [mm] diameter. The omnidirectional mobile robot of the main control parts are a control computer and motor systems. The control computer with I/O interfaces is set on a working table, and it connected with the robot by a long cable. The computer hard-ware is composed of D/A and counter cards. Three independent spherical-



**Figure 1:** The configuration of the single spherical-wheel assembly



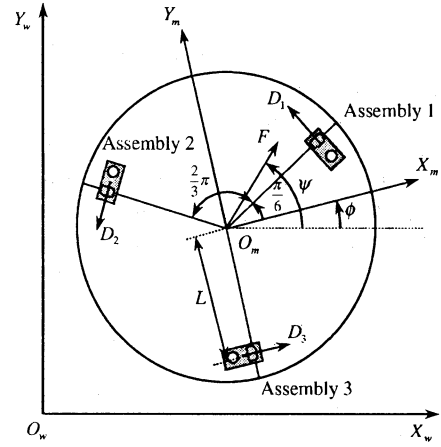
**Figure 2:** Omnidirectional mobile robot

wheels are driven by three servo-motors, where each of them has an optical encoder with 100 [Pulse/Rev]. The data from the encoder are fed back to the computer to perform the dead-reckoning.

Let the mobile robot be rigid moving on the work space. It is assumed that the absolute coordinate system  $O_w - X_w Y_w$  is fixed on the plane and the moving coordinate system  $O_m - X_m Y_m$  is fixed on the c.g. for the mobile robot as shown in Fig. 3. The constrained directions of motion of each assembly are indicated by the arrows labeled 1, 2 and 3.  $F$  is the force vector in the absolute coordinate system applied to the center of gravity for the mobile robot and  $\phi$  denote the angle between  $X_w$ - and  $X_m$ -coordinates, i.e., the rotational angle of the moving coordinate system with respect to the absolute coordinate system. Thus, defining the state variable for the robot as  $\mathbf{x} = [x_w \ y_w \ \phi \ \dot{x}_w \ \dot{y}_w \ \dot{\phi}]^T$ , the manipulated variable as  $\mathbf{u} = [u_1 \ u_2 \ u_3]^T$ , and the output variable as  $\mathbf{y} = [\dot{x}_w \ \dot{y}_w \ \phi]^T$  yields the following state equation:

$$\dot{\mathbf{x}} = A(\mathbf{x})\mathbf{x} + B(\mathbf{x})\mathbf{u} \quad (1)$$

$$\mathbf{y} = C\mathbf{x} \quad (2)$$



**Figure 3:** The model of omnidirectional mobile robot

where

$$A(\mathbf{x}) = \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} & 0 \\ 0_{3 \times 3} & a_1 & -a_2 \phi & 0 \\ & a_2 \phi & a_1 & 0 \\ & 0 & 0 & a_3 \end{bmatrix}$$

$$B(\mathbf{x}) = \begin{bmatrix} 0_{3 \times 3} & & \\ b_1 \beta_1 & b_1 \beta_2 & 2b_1 \cos \phi \\ b_1 \beta_3 & b_1 \beta_4 & 2b_1 \sin \phi \\ b_2 & b_2 & b_2 \end{bmatrix}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$a_1 = -3c/(3I_w + 2Mr^2), \quad a_2 = 3I_w/(3I_w + 2Mr^2)$$

$$a_3 = -3cL^2/(3I_w L^2 + I_v r^2)$$

$$b_1 = kr/(3I_w + 2Mr^2), \quad b_2 = krL/(3I_w L^2 + I_v r^2)$$

$$\beta_1 = -\sqrt{3} \sin \phi - \cos \phi, \quad \beta_2 = \sqrt{3} \sin \phi - \cos \phi$$

$$\beta_3 = \sqrt{3} \cos \phi - \sin \phi, \quad \beta_4 = -\sqrt{3} \cos \phi - \sin \phi$$

Let the transnational velocity of the robot in the absolute coordinate system be  $V = (\dot{x}_w^2 + \dot{y}_w^2)^{1/2}$  and the azimuth of the robot in the absolute coordinate system be  $\psi = \theta + \phi$ . Here,  $\theta$  denotes the angle between  $X_m$ -coordinate and  $f_m$ , i.e., the azimuth of the robot in the moving coordinate system. Then, it is found that

$$\dot{x}_w = V \cos \psi \quad (3)$$

$$\dot{y}_w = V \sin \psi \quad (4)$$

$$\psi = \arctan \frac{\dot{y}_w}{\dot{x}_w} \quad (5)$$

where note that a counter clockwise rotation denotes the positive direction for the rotational motion of the robot.

### 3. Fuzzy models



As shown the above dynamic model of the robot, it is considered that the nonlinearly of the dynamic system is mainly dominated by the rotational angle  $\phi$  of the robot between the moving and absolute coordinate systems. In order to build the model-based fuzzy servo system, we choose some representative rotational angles to simplify the nonlinear model to multiple linear models. Thus, we assumed four kind of models within the range of  $-2\pi \leq \phi \leq 2\pi$ , each fuzzy model rule is given as follows.

1. If  $\phi$  is about 0 or  $2\pi$  or  $-2\pi$  then
$$\Delta x(k+1) = A_d \Delta x(k) + B_{d1} \Delta u(k)$$

$$e(k) = e(k-1) - C \Delta x(k)$$
2. If  $\phi$  is about  $\pi/2$  or  $-3\pi/2$  then
$$\Delta x(k+1) = A_d \Delta x(k) + B_{d2} \Delta u(k)$$

$$e(k) = e(k-1) - C \Delta x(k)$$
3. If  $\phi$  is about  $\pi$  or  $-\pi$  then
$$\Delta x(k+1) = A_d \Delta x(k) + B_{d3} \Delta u(k)$$

$$e(k) = e(k-1) - C \Delta x(k)$$
4. If  $\phi$  is about  $3\pi/2$  or  $-\pi/2$  then
$$\Delta x(k+1) = A_d \Delta x(k) + B_{d4} \Delta u(k)$$

$$e(k) = e(k-1) - C \Delta x(k)$$

where  $\Delta(z^{-1}) = 1 - z^{-1}$ ,  $z^{-1}$  denotes the one-step delay operator.  $A_d$  and  $B_{di}$  are the system and control distribution matrices in discrete-time, respectively. Then, the reference output error is defined by

$$e(k) \triangleq y_d - y(k) \quad (6)$$

where the reference output vector  $y_d = [\dot{x}_{wd} \ \dot{y}_{wd} \ \phi_d]^T$ .

Let the new state variable is to be  $[\Delta x(k+1), e(k)]^T$ , the augmented system of Eq. (1) can be described as

$$\begin{bmatrix} \Delta x(k+1) \\ e(k) \end{bmatrix} = \Phi_d \begin{bmatrix} \Delta x(k) \\ e(k-1) \end{bmatrix} + \Gamma_{di} \Delta u(k) \quad (7)$$

where  $\Phi_d$  and  $\Gamma_{di}$  are the system matrix and the control distribution matrix for the augmented system described by

$$\Phi_d = \begin{bmatrix} A_d & 0_{6 \times 3} \\ -C & I_{3 \times 3} \end{bmatrix}; \quad \Gamma_{di} = \begin{bmatrix} B_{di} \\ 0_{3 \times 3} \end{bmatrix}$$

Thus, the manipulated quantity of this system can be written as

$$u(k) = -F_i x(k) - K_i \Delta^{-1} e(k-1) \quad (8)$$

where the optimal servo gain matrices are denoted by

$$G_i = [R_i + \Gamma_{di}^T P_i \Gamma_{di}]^{-1} \Gamma_{di}^T P_i \Phi_d$$

$$\triangleq [F_i, K_i] \quad (9)$$

and defining  $\Delta^{-1} e(k-1) = z(k-1)$ , its manipulated quantity can become

$$u(k) = -F_i x(k) - K_i z(k-1) \quad (10)$$

Note that in this case, it is difficult to obtain a unique stabilization solution  $P_i > 0$ . But, since there

exist uncontrollable model on the unit circle, it is possible to obtain  $P_i \geq 0$  as a strong solution by applying an iterative approach.

The membership function is used a Gaussian function type with the center value  $\omega_c$ , whose confidence  $\mu$  is given by

$$\mu = \exp\{\ln(0.5)(x - \omega_c)^2 \omega_d^2\} \quad (11)$$

where the input variable is  $x$  and the reciprocal value of standard deviation is  $\omega_d$ . Since there exist two or three multiple labels in one model rule, the confidence of  $i$ -th model rule in the antecedent part is assumed to be given by

$$h_i = \max_j \{\mu_{ij}\} \quad (12)$$

where  $j$  denotes the number of fuzzy labels in the  $i$ -th model rule. When defining the normalized confidence given by

$$p_i = h_i / \sum_{j=1}^4 h_j \quad (13)$$

The manipulated quantity can be consequently expressed by

$$u(k) = - \left[ \sum_{i=1}^4 F_i p_i \right] x(k) - \left[ \sum_{i=1}^4 K_i p_i \right] z(k-1) \quad (14)$$

The block diagram of type-1 fuzzy servo control system is shown in Fig. 4.

## 4. Experimental Results

In this section, some experiments are conducted to demonstrate the performance of above method. The physical parameters of the mobile robot are as follows:

$$I_v = 19.81 \text{ [kgm}^2\text{]}, \quad M = 8.7 \text{ [kg]},$$

$$L = 0.1775 \text{ [m]}, \quad k = 0.180, \quad c = 0.6965 \times 10^{-6} \text{ [kgm}^2\text{/s]},$$

$$I_w = 0.0219 \text{ [kgm}^2\text{]}, \quad r = 0.03 \text{ [m]}$$

It was assumed that the initial moving coordinate system is equal to the absolute coordinate system, and the initial value of state variable is given as  $x = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T$ .

In this case, the experiment for a triangular line path was considered with an experimental time 15 [s], in which the desired velocity  $v_d$  was given by 0.06 [m/s] and the sampling period was 100 [ms]. The desired azimuth  $\psi_d$  is set up  $\pi/3$  [rad] (i.e.,  $60^\circ$ ) for the initial time duration 5.0 [s]. Then, the robot was turned to the left with  $\psi_d = 2\pi/3$  [rad] ( $120^\circ$ ), and it is again turned to  $\psi_d = 120^\circ$  for the final time duration 5.0 [s]. Here, note that  $\phi_d \equiv 0$ . The experimental result in the  $(x, y)$ -coordinate is given by Fig. 5. From this figure, it is seen that a good trajectory tracking is obtained.

Next, a straight line path was considered with a varying rotational angle, whose change is from 0 [rad] to  $\pi/3$  [rad] in the twinkling of the beginning control. The trajectory was assumed to be a constant velocity  $v_d = 0.1$  [m/s] and  $\psi_d = \pi/6$  [rad]. The diagonal matrices  $Q$  and  $R$  for the Riccati equation were found as

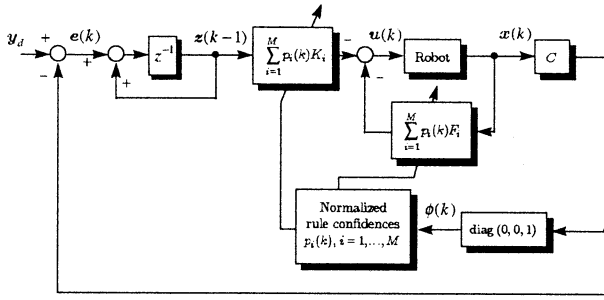


Figure 4:Block diagram for type-1 fuzzy servo system

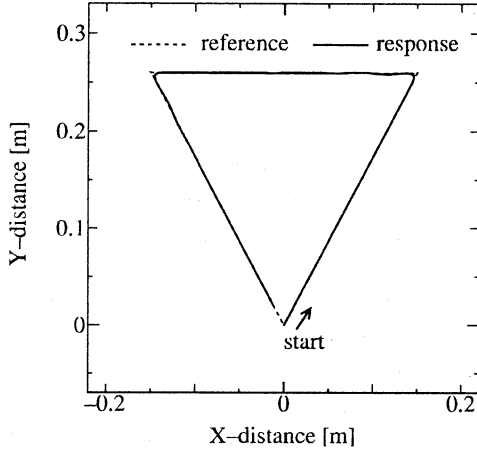


Figure 5:Trajectory control results of the triangular path

$Q_i = \text{diag}(20, 20, 20, 100, 100, 100, 400, 400, 226)$ ,  $R_i = \text{diag}(1, 1, 1)$ . Figure 6 shows the responses of the X-,Y- velocity, and the response of the rotational angle is also shown in Fig. 7. Figure 8 shows the normalized confidence for each model rule. From this figure, it is seen that 1/3 part of the model rule 1 and 3/4 part of the model rule 2 were used.

## 5. Conclusions

We adopt three single spherical-wheel assemblies consisted of an omnidirectional holonomic mobile robot. We have also proposed a fuzzy servo controller based on using several fuzzy models. Several experimental results produced that the robot gives the desirable full mobility and effective for controlling this mobile robot.

## References

- [1] F. G. Pin and S. M. Killough, "A New Family of Omnidirectional and Holonomic Wheeled Platforms for Mobile Robots," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 4, pp. 480-489, 1994.
- [2] J. Tang and K. Watanabe, "Design of Optimal Servo Systems Based on Fuzzy Models for an Omnidirectional Mobile Robot," *J. of JSME*, Vol.64C, No.626, pp. 202-208, 1998 (in Japanese).

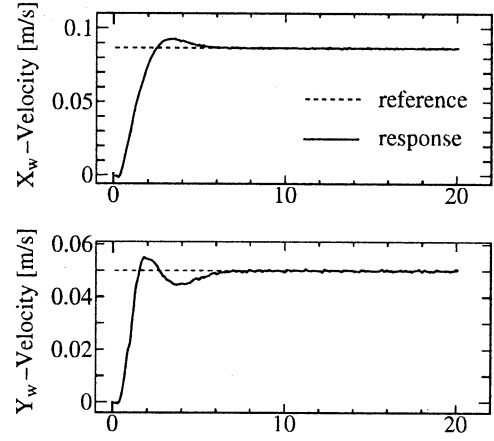


Figure 6: Velocity control results

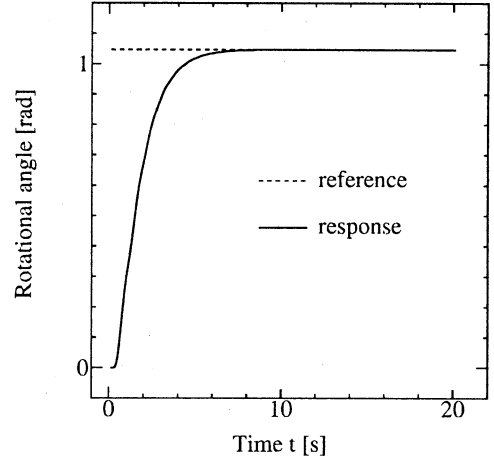


Figure 7:Rotational angle control for the straight path

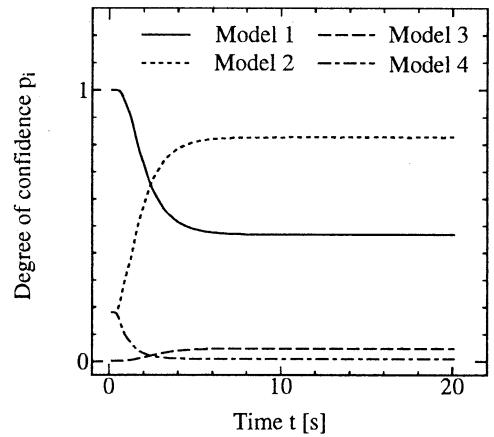


Figure 8:Normalized confidences for the straight path

## An Experiment on Force Control Using Fuzzy Environment Models

Fusaomi Nagata\*, Keigo Watanabe\*\*, Kiyotaka Izumi\*\*\*, Kazuya Sato\*\* and Sinobu Akama†

\*Interior Design Research Institute, Fukuoka Industrial Technology Center,  
Agemaki-405-3, Ohkawa, Fukuoka 831-0031, JAPAN  
nagata@idri01.fitc.pref.fukuoka.jp

\*\*Department of Advanced Systems Control Engineering, Graduate School of Science and Engineering,  
Saga University, Honjomachi-1, Saga 840-8502, JAPAN

\*\*\*Department of Mechanical Engineering, Faculty of Science and Engineering,  
Saga University, Honjomachi-1, Saga 840-8502, JAPAN

†Department of Mechanical Engineering, Graduate School of Science and Engineering,  
Saga University, Honjomachi-1, Saga 840-8502, JAPAN

### Abstract

We have already proposed a position-based impedance control using fuzzy environment models (FEMs) learned with genetic algorithms so that industrial robots can realize a stable force control for unknown environments. The models estimate the stiffness of the unknown environments and yield the desired damping to suppress overshoots and oscillations considering the critical damping condition with the environments. In the present paper, the proposed method is applied to an industrial robot Js-10 with an open architecture controller, and the effectiveness and promise are proved through a force control experiment.

**Key Words:** Impedance Control, Fuzzy Environment Models, Genetic Algorithms, Industrial Robot, Open Architecture Controller, Unknown Environment

### 1. Introduction

Impedance control [1] is one of the most effective strategies for a manipulator to desirably contact with an environment. It is globally expected that the manipulator incorporating the impedance control will be applied to various tasks which need compliant force control just like being easily performed by human being. This method is characterized by an ability of controlling the mechanical impedance acting between the end-effector and environment, such as an inertia, damping and stiffness.

Among these parameters, the damping is the most important to realize a stable force control [2, 3]. When the conventional impedance control methods are applied, we must adjust the damping according to the each task and environment with trial and error. In this case, it is necessary to know the stiffness of each environment in advance. To automate the complicated tuning of the damping, we have already proposed the FEMs [4, 5] that is learned with genetic algorithms. The simulation results have shown that the FEMs can estimate the stiffness of the unknown environment and yield the desired time-varying damping for the desirable force control.

In this paper, we first review the position-based impedance control for industrial robots and the tuning method of desired damping considering the environmental dynamics. Next, FEMs used in the experiment are explained in detail. Finally, we implement the position-based impedance control using the FEMs in an industrial robot Js-10 provided by Kawasaki Heavy Industries. The robot is applied to the force control experiment. The effectiveness of the present approach is examined by comparing the two cases, i.e. without FEMs and with FEMs.

### 2. Position-based impedance control

The desired impedance equation of a robot manipulator is designed by

$$\begin{aligned} M_x(\theta)\ddot{x} + B_d(\dot{x} - \dot{x}_d) + K_d(x - x_d) \\ = SF + (E - S)K_f(F - F_d) \end{aligned} \quad (1)$$

where  $x$ ,  $\dot{x}$  and  $\ddot{x}$  are the position, velocity, and acceleration vectors, respectively.  $M_x(\theta)$  is the inertia matrix,  $F$  is the force-moment vector acting on the end-effector defined by  $F^T = [f^T n^T]$ , where  $f$  and  $n$  are the force and moment vectors, respectively.  $K_f$  is the force feedback gain matrix.  $x_d$ ,  $\dot{x}_d$  and  $F_d^T = [f_d^T n_d^T]$  are the desired position, velocity and force vectors;  $B_d$  and  $K_d$  are the coefficient matrices of the desired damping and stiffness, respectively.  $S$  and  $E$  are the switch matrix  $\text{diag}(S_1, \dots, S_6)$  and identity matrix. It is assumed that  $K_f$ ,  $B_d$  and  $K_d$  are positive-definite diagonal matrices. It should be noted that eq. (1) has no switch matrix  $S$  applied to the desired stiffness in Hybrid Compliance/Force Control (HCC) [2]. We consider here a position control strategy in the direction for force control, if the object dynamics is known. Note that if  $S = E$ , eq. (1) becomes a compliance control system in all directions; whereas if  $S = \mathbf{0}$ , it becomes a force control system in all directions.

It is assumed that the displacement of eq. (1) is very small and the inertia term can be ignored. Defining

$\mathbf{x}(k)=\mathbf{x}(t)|_{t=\Delta t k}$  ( $k=0,1,\dots$ ), the recursive equation of position command in the Cartesian space can be given by [3]

$$\mathbf{x}(k) = \mathbf{x}_d(k) + e^{-B_d^{-1} K_d \Delta t} \left\{ \mathbf{x}(k-1) - \mathbf{x}_d(k-1) \right\} - \left\{ e^{-B_d^{-1} K_d \Delta t} - E \right\} K_d^{-1} \left\{ S\mathbf{F} + (E-S)K_f(F-F_d) \right\} \quad (2)$$

### 3. A tuning method considering the dynamics of environments

Figure 1 shows that the end-effector is made to contact with an environment tilt by  $\theta$  degrees and it is tried to control the contact force from the vertical direction of the slope to converge to the reference  ${}^S f_d$ . Here, the superscript  $S$  means the value in sensor coordinate system. In this case, the rotation matrix  ${}^B R_s$  in sensor coordinate system based on base coordinate system is expressed by

$${}^B R_s = \begin{pmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{pmatrix} \quad (3)$$

So, the desired force vector  $f_d$  in base coordinate system is obtained by

$$f_d = {}^B R_s {}^S f_d \quad (4)$$

For example,  ${}^S f_d = [0 \ 0 \ {}^S f_{dz}]^T$  [N] is transformed to  $f_d = [-{}^S f_{dz} \sin(\theta) \ 0 \ {}^S f_{dz} \cos(\theta)]^T$  [N]. Consider a situation where the contact force  $F$  acting at the end-effector is represented by

$$F = -B_m \dot{x} - K_m (x - x_m) \quad (5)$$

where  $B_m$  [Ns/m] and  $K_m$  [N/m] are the viscosity and stiffness coefficients to be positive-definite diagonal matrices, and  $x_m^T = [x_m \ y_m \ z_m \ \phi_m \ \theta_m \ \psi_m]$  is the position of the object. The desired damping  $\tilde{B}_{di}$  considering the critical

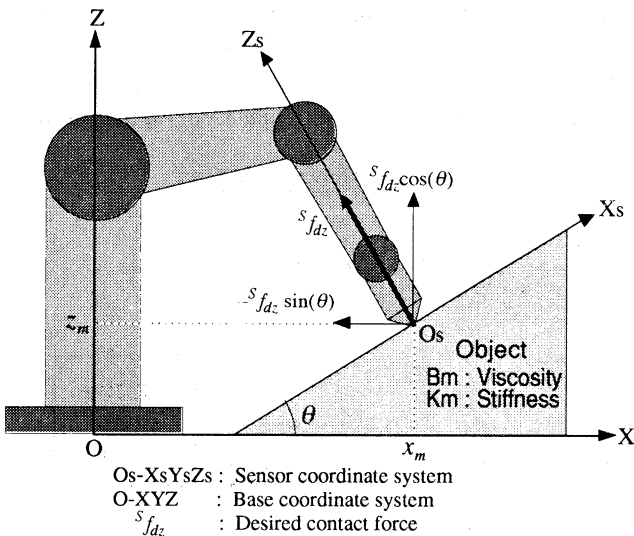


Fig. 1 A force control situation used in the experiment

damping condition [3] of eq. (1) is given by

$$\tilde{B}_{di} = 2\sqrt{\left(J^T M J^{-1}\right)_i \left[K_{di} + \left\{S_i K_{mi} + (E-S)_i K_f K_{mi}\right\} - \left\{S_i B_{mi} + (E-S)_i K_f B_{mi}\right\}\right]} \quad (6)$$

Here,  $i$  ( $i=1,\dots,6$ ) denotes the  $i$ -th diagonal element of each matrix and  $M$  is the inertia term in the joint space. In our proposed position-based impedance control,  $\tilde{B}_{di}$  is used as a reference of desired damping so that the sufficient performance for a stable force control can be realized. When using this method, the physical parameters of the environment must be known to calculate  $\tilde{B}_{di}$ .

### 4. Fuzzy environment models (FEMs)

It is known that the response of force control is seriously affected by the stiffness of environment and the viscosity has little influence on the stability. From this fact, it is enough to know the stiffness of environment to realize a stable force control without overshoots and oscillations. Therefore, we propose the FEMs that can estimate the stiffness of unknown environment.

The FEMs estimate the stiffness of environment using the simple reasoning method with single input/output. In this section, it is described on how to construct the FEMs, giving an example of force control situation shown in Fig. 1. The position vector of the end-effector and the force vector acting at the end-effector are  $f^T = [f_x(t) \ 0 \ f_z(t)]$  and  $x^T = [x(t) \ y(t) \ z(t) \ \phi(t) \ \theta(t) \ \psi(t)]$  respectively. Each reasoning in the  $x$ - or  $z$ -direction is conducted independently. It is assumed that each input for reasoning is written by

$$x^*(t) \triangleq f_x(t) / (x_m - x(t)) \quad (7)$$

$$z^*(t) \triangleq f_z(t) / (z_m - z(t)) \quad (8)$$

In this case, the fuzzy rules in the  $x$ -direction are composed by

$$\begin{aligned} \text{Rule 1: IF } x^*(t) \text{ is } \tilde{A}_1 \\ \text{THEN } B_m^1 \dot{x} + K_m^1 (x - x_m) &= -F \\ \text{Rule 2: IF } x^*(t) \text{ is } \tilde{A}_2 \\ \text{THEN } B_m^2 \dot{x} + K_m^2 (x - x_m) &= -F \\ &\vdots \\ \text{Rule } L: \text{ IF } x^*(t) \text{ is } \tilde{A}_L \\ \text{THEN } B_m^L \dot{x} + K_m^L (x - x_m) &= -F \end{aligned}$$

where  $\tilde{A}_i$  ( $i=1,\dots,L$ ) is the antecedent fuzzy set,  $L$  is the label number. If  $K_{m1}$  to be  $x$ -directional element of stiffness coefficient matrix  $K_m$  is estimated in parallel with these fuzzy rules, the reasoning rules can be obtained by

Rule 1: IF  $x^*(t)$  is  $\tilde{A}_1$   
 THEN  $K_{m1} = K_{m11}$   
 Rule 2: IF  $x^*(t)$  is  $\tilde{A}_2$   
 THEN  $K_{m1} = K_{m12}$   
 $\vdots$   
 Rule  $L$ : IF  $x^*(t)$  is  $\tilde{A}_L$   
 THEN  $K_{m1} = K_{m1L}$

where  $K_{m1i}$  ( $i=1, \dots, L$ ) denotes the consequent constant describing the stiffness concerning the rule  $i$ . Using the above rules, the estimate value  $\hat{K}_{m1}$  in the  $x$ -direction is computed by the weighted mean method expressed as

$$\hat{K}_{m1} = \sum_{i=1}^L p_i(x^*(t)) K_{m1i} \quad (9)$$

where  $p_i(x^*(t))$  denotes the normalized fitness given by

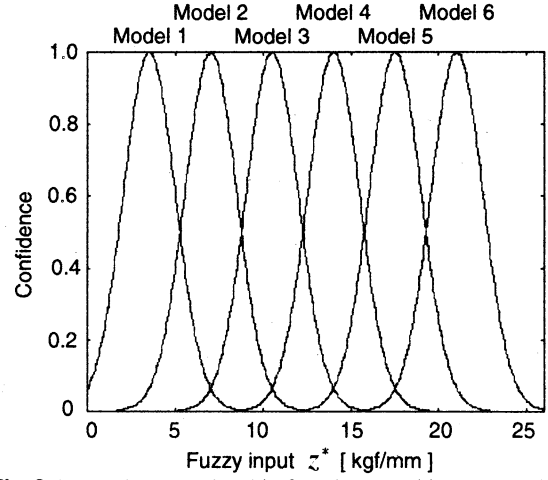
$$p_i(x^*(t)) = \frac{\mu_{\tilde{A}_i}(x^*(t))}{\sum_{j=1}^L \mu_{\tilde{A}_j}(x^*(t))} \quad (10)$$

with the antecedent fitness  $\mu_{\tilde{A}_j}(x^*(t))$  described by the following Gaussian membership function

$$\mu_{\tilde{A}_j}(x^*(t)) = \exp\left\{\log(0.5)(x^*(t) - \alpha_j)^2 \beta_j^2\right\} \quad (11)$$

in which  $\alpha_j$  is the center of membership function,  $\beta_j$  is the reciprocal value of standard deviation,  $j$  ( $j = 1, \dots, L$ ) denotes the number of fuzzy rules. In the same manner, the estimate value  $\hat{K}_{m3}$  in the  $z$ -direction can be computed by giving eq. (8) to the input of fuzzy reasoning.

We have previously designed the FEMs with generalization. The generalized FEMs are obtained by integrating the FEMs learned under several known environments using genetic algorithms [4, 5]. Before the following experiment, we adjust the two points. The one is support set range of antecedent part according to the actual fuzzy inputs. The other is the scale of consequent constants which are expressing the estimated stiffness of each model. **Figure 2** and **Table 1** are the adjusted



**Fig. 2** Antecedent membership functions used in the experiments

**Table 1** Consequent constants used in the experiments

Model 1	Model 2	Model 3	Model 4	Model 5	Model 6
0.5876	3.4824	8.7539	17.6555	24.0366	38.2986

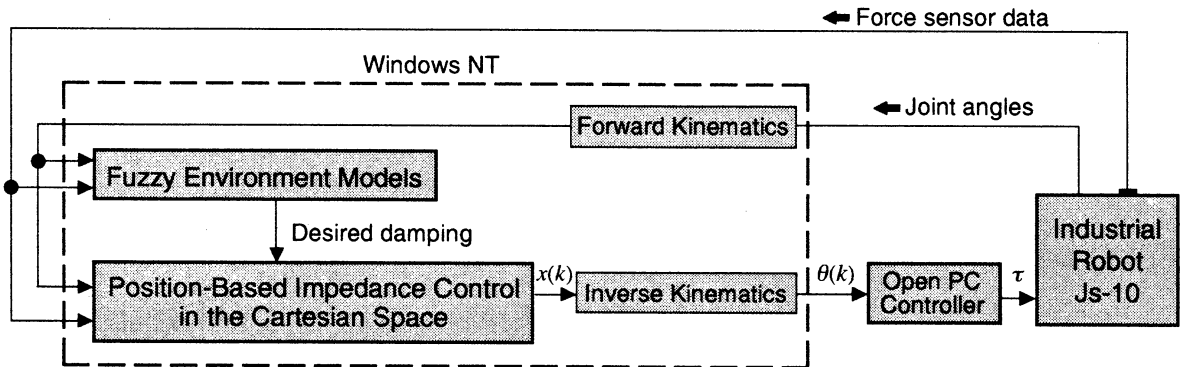
antecedent membership functions and consequent constants.

## 5. An experiment of one-dimensional force control

**Figure 3** shows the block diagram of position-based impedance control using the generalized FEMs. The FEMs estimate the stiffness of environment using the force sensor measurements and yield the desired damping  $\tilde{B}_d$  which can maintain the system in the critical damping. The position-based impedance control generates the reference  $x(k)$  for the servo system with  $\tilde{B}_d$ . In the following, we show some experimental results using an industrial robot Js-10 with an open architecture controller. **Figure 4** shows the experimental scene of  $z$ -directional force control, in which the cases of not using FEMs and using FEMs are compared.

### 5.1 In the case of not using FEMs

In this experiment, we made the end-effector approach to a wood with constant velocity 1 [mm/s] and come in contact with the wood with 3 [kgf]. As a first case study, the desired damping was selected with trial and error. Of course, the FEMs were not used. **Figure 5** shows



**Fig. 3** Block diagram of the experimental system

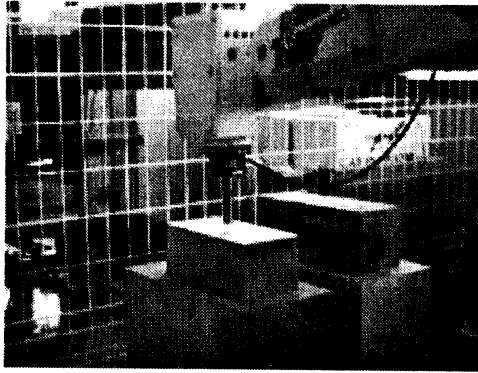


Fig. 4 A force control experiment in the z-direction

the experimental results of force control with several constant desired dampings. Consequently, it has been confirmed that the response becomes underdamped if the desired damping is too small, on the other hand the response becomes overdamped if it is too large. It can be seen from Fig. 5 that we must find out the suitable damping to realize a stable force control like (b).

### 5.2 In the case of using FEMs

Another case study was carried out to investigate the ability of the FEMs when the contact occurs with three unknown objects. The objects are a metal, normal wood and soft wood. The force control results are presented in Fig. 6. The time histories of desired dampings are also shown in Fig. 7. It can be found that the desirable responses, in which overshoots and oscillations are suppressed, are obtained for any objects.

## 6. Conclusions

In this study, the effectiveness of fuzzy environment models has been examined through a force control experiment using an industrial robot Js-10 with an open architecture controller. Consequently, it has been recognized that the fuzzy environment models offer a good force control performance in the case of uncertainties on the stiffness of environments.

## References

- [1] N. Hogan: Impedance control: "An Approach to Manipulation: Part I - Part III," *Trans. ASME Journal of Dynamic Systems, Measurement and Control*, vol. 107, pp. 1 - 24, 1985
- [2] T. Katsuragawa, K. Ioi, N. Kubota and O. Noro: "Application of Hybrid Compliance/Force Control for Industrial Robot," *Journal of the Robotics Society of Japan*, vol. 12, no. 6, pp. 893-898, 1994 (in Japanese)
- [3] F. Nagata, K. Watanabe, K. Sato, K. Izumi and T. Suehiro: "Impedance Control for Articulated Robot of 6 Degree-of-Freedom in Consideration of Critically Damped Condition with an Object Dynamics," *Proc. of the 36th SICE Annual Conference*, pp. 1119-1124, 1997

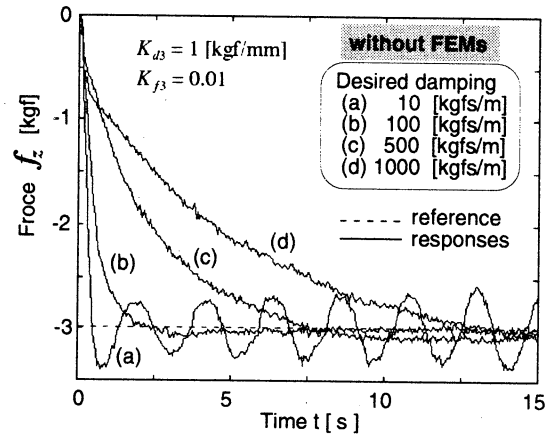


Fig. 5 z-directional forces using manually tuned dampings

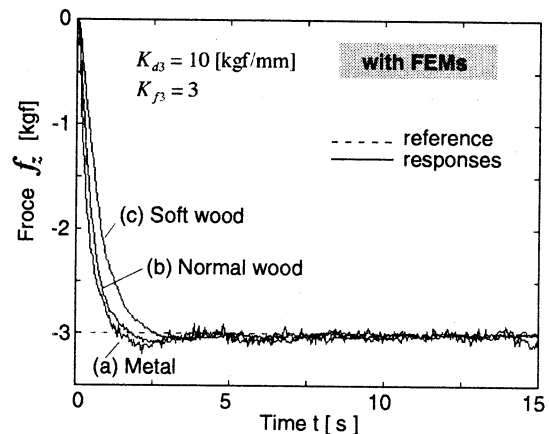


Fig. 6 z-directional forces using fuzzy environment models

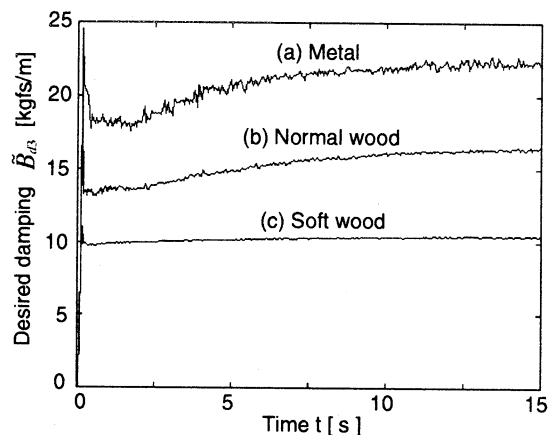


Fig. 7 z-directional desired damping  $\tilde{B}_{d3}$  generated by FEMs

- [4] F. Nagata, K. Watanabe, K. Sato and K. Izumi: "Position-Based Impedance Control with Fuzzy Environment Models," *Procs. of the 37th SICE Annual Conference, International Sessions*, pp. 837-842, 1998
- [5] F. Nagata, K. Watanabe, K. Sato and K. Izumi: "An Application for Polishing Using Impedance Control with Fuzzy Environment Models," *Procs. of the 4th Japan-France Congress & 2nd Asia-Europe Congress on Mechatronics*, pp. 237-242, 1998

## Realization of Artificial Human Decision Making Based on Conditional Probability

Masatoshi NAKAMURA, Satoru GOTO and Takenao SUGI

Department of Advanced Systems Control Engineering,

Graduate School of Science and Engineering,

Saga University

Honjomachi, Saga 840-8502, Japan

### Abstract

Procedure of a qualified human decision making includes many intuitive factors which have been acquired from one's valuable experiences and gained through one's learning, but is not easy understandable to others within short term. By use of the database of causes and decision results done by qualified experts for an objective event, artificial human decision making for the event will be realizable. This paper investigates a general method for realizing artificial human's on-off decision making based on conditional probability of the database. As the on-off decision making is an discrete event and the causes for the decision making are continuous events, a mathematical care for a treatment of Dirac delta function in the probability density function is required to derive the conditional probability for the decision making. Several examples of artificial human decision makings by the proposed method were demonstrated with satisfactory evaluations.

**Key words** On-off human decision making, Conditional probability, Insulator washing timing, Spike detection

### 1 Introduction

Many on-off human decision making problems exist in our daily life, industries and medical fields so on. Examples of the problem can be found in a judgment of medical doctors for the disease of subjects. Visual inspection of spikes on EEG record for a subject of epilepsy is one of the problems. Medical doctors inspect the EEG time series and try to find out specific waves which represent significant features of the spike on the EEG record[2]. Similar problems can be found in industries. Experts in a power substation must make decision for washing out polluted deposits from insulators to prevent flash over due to the deterioration of insulation[1]. Another example is that a technical manager makes maintenance scheduling of equipments in the factory to circumvent failures. Many researchers and technicians have been engaged in automatization of these decision makings. Although much efforts have been done for automatization, final automatized techniques required some threshold values for the decision making and determined by something like ad hoc ones.

Then, logical and reasonable automatic realization of on-off human decision procedure is required to develop the automatic methods.

This paper investigates an analysis of on-off human decision making process based on conditional probability of database and proposes a reasonable procedure for artificial realization of on-off human decision making. Several successful examples of the artificial human decision makings were demonstrated.

## 2 Artificial realization of on-off human decision making

### 2.1 Problem statement

Conceptual explanation of artificial realization of on-off human decision making is investigated. Based on database of the on-off decision making Go (yes) or Nogo (no) done by experts under the respective circumstances, automatic on-off decision making procedure is developed by use of conditional probability of the database.

Acquired data set  $\{y\}$  is classified into two groups of human decision  $x$  (1: Go, 0: Nogo). The measurement  $y$  consists of several elements and is represented by a vector variable. Numbers of the decision making for Go are  $M$ , and those for Nogo are  $N$ , respectively. The problem of artificial realization of on-off human decision making is reduced to obtain the probability  $Go P(1|y)$  based on the current measurement  $y$ .

### 2.2 Automatic on-off decision making

**Frequency and histogram** Based on  $M + N$  data set of  $\{x, y\}$ , frequencies of Go and Nogo and the histograms of  $y$  for Go and Nogo are calculated as follows. The probability of Go  $P(1)$  and that of Nogo  $P(0)$  are defined as the relative frequencies for the discrete events Go and Nogo as follows:

$$P(1) = M/(M + N), \quad P(0) = N/(M + N). \quad (1)$$

The normalized histogram of the measurement data  $y$  for Go and that for Nogo are denoted by  $h(y|1)$  and  $h(y|0)$  as seen in Fig. 1-I. The histograms have the following property of the normalization as

$$\int h(y|1)dy = 1, \quad \int h(y|0)dy = 1. \quad (2)$$

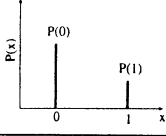
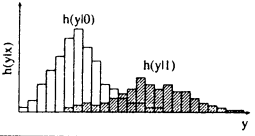
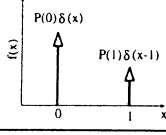
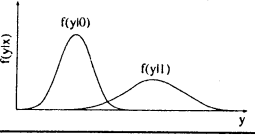
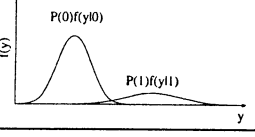
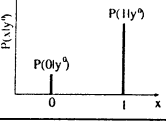
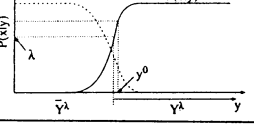
		Go and Nogo (Descrete Event)	Acquisited Data (Continuous Event)
I	Freq. and Hitogram		
II	Pdf		
III	Cond. Prob.	$P(1 y)$	
IV	Prob. of Go		

Figure 1: Procedure of automatic realization of on-off decision making based on a conditional probability.

**Probability density function** To obtain the probability of Go  $P(1|y)$  based on the measurement data  $y$ , we use Bayesian rule[7] which requires the probability density function  $f(x)$  for  $x$  and that of  $f(y|x)$  conditioned on  $x$ .

The probability density function of  $f(x)$  for discrete event  $x$  as seen in Fig. 1-II is derived by use of the probability  $P(0)$  and  $P(1)$  and Dirac delta function  $\delta(x)$  as

$$f(x) = P(0)\delta(x) + P(1)\delta(x-1). \quad (3)$$

The probability density functions of  $f(y|1)$  and  $f(y|0)$  are obtained by approximating  $h(y|1)$  and  $h(y|0)$  under the restriction of the probability density function as

$$\int f(y|1)dy = 1, \quad \int f(y|0)dy = 1. \quad (4)$$

The least squares method can be used to derive the probability density function  $f(y|1)$  and  $f(y|0)$ . Fig. 1-II illustrates the probability density functions for discrete event  $x$  and that for continuous event  $y$ .

**Bayesian rule** By use of Bayesian rule and (3), we obtain  $f(x|y)$  as

$$f(x|y) = \frac{f(x, y)}{f(y)} = \frac{f(x, y)}{\int f(x, y)dx} = \frac{f(x)f(y|x)}{\int f(x)f(y|x)dx}$$

$$= \frac{P(0)f(y|x)\delta(x) + P(1)f(y|x)\delta(x-1)}{P(0)f(y|0) + P(1)f(y|1)}. \quad (5)$$

In the derivation of the denominator of (5), integral properties of the generalized function[6] of Dirac delta function are adopted as

$$\begin{aligned} \int P(0)f(y|x)\delta(x)dx &= P(0)f(y|0), \\ \int P(1)f(y|x)\delta(x-1)dx &= P(1)f(y|1). \end{aligned} \quad (6)$$

**Conditional probability** Conditional probability for Go  $P(1|y)$  can be calculated by taking the integral of the impulsive conditional probability density function (5) around the neighborhood of 1(Go) as follows:

$$\begin{aligned} P(1|y) &= \int_{1-\epsilon}^{1+\epsilon} f(x|y)dx \\ &= \frac{P(1)f(y|1)}{P(0)f(y|0) + P(1)f(y|1)}. \end{aligned} \quad (7)$$

Fig. 1-III shows the each term in (7). The summation of the two terms shows the probability density function of the data  $y$  by comparing the both denominations of (7) and (5) as

$$f(y) = P(0)f(y|0) + P(1)f(y|1). \quad (8)$$



Fig. 1-IV illustrates the conditional probability  $P(1|\mathbf{y})$  based on the data  $\mathbf{y}$ . The conditional probability of Nogo  $P(0|\mathbf{y})$  can be obtained by taking a similar procedure for the calculation of  $P(1|\mathbf{y})$ . In the latter case, the integral of  $f(x|\mathbf{y})$  is taken around the neighborhood of 0 (Nogo) as

$$\begin{aligned} P(0|\mathbf{y}) &= \int_{-\epsilon}^{\epsilon} f(x|\mathbf{y})dx \\ &= \frac{P(0)f(\mathbf{y}|0)}{P(0)f(\mathbf{y}|0) + P(1)f(\mathbf{y}|1)}. \end{aligned} \quad (9)$$

**On-off decision making** Automatic decision making of Go is done if the conditional probability  $P(1|\mathbf{y}^0)$  for a given measurement data  $\mathbf{y}^0$  is greater than or equal to a given threshold value  $\lambda$  as

$$P(1|\mathbf{y}^0) \geq \lambda \rightarrow \text{Go}. \quad (10)$$

Equation (10) shows a criteria for artificial realization of the decision making for a given value of  $\lambda$ . The value  $\lambda$  shows the rate of Go for the database obtained from the experts. The value 1 for  $\lambda$  represents that the expert always made the decision of Go under the circumstances of the measurement  $\mathbf{y}^0$ . The value  $\lambda$  should be determined by the designer for the automatic decision making depending on the designers' desire.

**Properties of the automatic on-off decision making** Accuracy of the decision making of Go, rate of the false negative, and rate of the false positive are calculated by  $\int_{Y^\lambda} f(\mathbf{y}|1)d\mathbf{y}$ ,  $\int_{\bar{Y}^\lambda} f(\mathbf{y}|1)d\mathbf{y}$ ,  $\int_{Y^\lambda} P(0)f(\mathbf{y}|0)\mathbf{y} / \int_{Y^\lambda} f(\mathbf{y})d\mathbf{y}$ , respectively, where  $Y^\lambda$  is the region defined by  $\{\mathbf{y} \in Y^\lambda | P(1|\mathbf{y}) \geq \lambda\}$  and  $\bar{Y}^\lambda$  is the counter set of  $Y^\lambda$ .

It is very effective for designer to be able to know the accuracy of the automatic decision making before the implementation of the decision making. These informations are usable for designer to select prominent variable of the measurement vector  $\mathbf{y}$  out of whole database. The appropriate selection of the measurement variable  $\mathbf{y}$  will be done in order to reduce the area of intersection of  $f(\mathbf{y}|0)$  and  $f(\mathbf{y}|1)$  as seen in Fig. 1-II. If the intersection of two functions is zero by appropriate selection of  $\mathbf{y}$ , the automatic decision always coincides with the human decision making.

### 3 Artificial human decision making for insulator washing timing

#### 3.1 Objective of insulator washing

Strong wind from seashore often causes rapid pollution of insulators in coastal substations. The rapid pollution of insulators caused by strong wind is very dangerous. The analysis of the actual data reveals that it takes only two or three hours to reach the hazardous levels of pollution in typhoons seasonal windy days

and attack of typhoons. The insulators are washed with much amount of pure water before the pollution deposits increase to a critical level. In actual practice, based on the decision making by human experts, the polluted insulators are washed[3].

#### 3.2 Automatic decision making of insulator washing timing

Automatic decision making of insulator washing timing was derived based on the artificial realization of on-off human decision making procedure described in Chapter 2. By use of the raw data of weather conditions and insulator pollution acquired from Karatsu substation in Japan during the period of April, September and October 1994, an artificial human decision making for insulator washing timing was constructed.

The non conditional probability for Go (washing) and that for Nogo (no washing) were

$$P(1) = 33/(923 + 33), \quad P(0) = 923/(923 + 33) \quad (11)$$

and the normalized histograms for pollution deposits  $y_1$ , wind velocity  $y_2$ , wind direction  $y_3$  and rainfall  $y_4$  are illustrated in Fig. 2. By use of the equations from (3) to (10), we calculated the automatic decision making of insulator washing timing.

#### 3.3 Evaluation of the decision making of insulator washing timing

The proposed decision making method was verified by using the actual collected data. In the actual data, the pilot insulator for the measurement of the pollution deposits, was also washed as same as the actual insulators. The modified data, which neglected the effect of the actual insulator washing, were constructed from the actual data for the validation of the proposed decision making method. The modified pollution deposits  $d_m(t)$  were calculated by following procedure.

1. The actual pollution deposit difference  $\Delta d(t)$  was added to the current modified pollution deposits  $d_m(t)$  for the pollution deposits in the next step if the insulator was not washed.
2. Drop of the pollution deposits due to insulator washing was neglected in the modified pollution deposits  $d_m(t)$ .
3. When the proposed method decided the insulator washing, the modified pollution deposits  $d_m(t)$  were set to the mean value of the pollution deposits, just after actual washings.
4. If the pollution deposits became less than zero, as a result of the combined effect of the modified washing and the natural washing, the modified pollution deposits  $d_m(t)$  were set to zero.

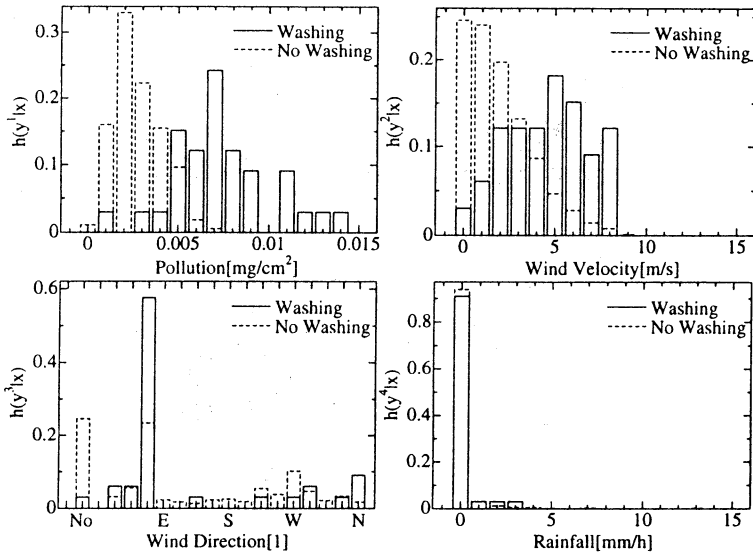


Figure 2: Normalized histograms of pollution deposit, wind velocity, wind direction and rain fall for Go(washing) and Nogo(no

1994 at Karatsu substation in Japan.

The results of the proposed decision making method in case of  $\lambda$  being 0.5 is shown in Fig. 3. The total number of insulator washings using the proposed method were 7 which was the same numbers as that of the actual washing and the washing timings determined by the automatic one was similar timings done by the practitioner in actual field. These results revealed that the artificial human decision making for insulator washing timing was accomplished successfully by the proposed conditional probability base.

## 4 Artificial human decision making for spike detection

### 4.1 Necessity of automatic spike detection

Spikes detected on electroencephalographic (EEG) records provide important informations for clinical diagnosis, especially in that of epileptic disorders. Usually, the spikes are detected by electroencephalographer (EEGer) with much laborious work. A criteria for spikes in visual inspection is defined as a wave whose shape has spike immediately followed by clear slow activity or clear spike outstanding well from the background activity as seen in Fig. 4. An automatic spike detection is, if correctly applied, a powerful aid for EEGers in the detection of spikes in the long-lasting EEG records such as for epilepsy monitoring.

### 4.2 Automatic spike detection

**Subjects and data acquisition** EEGs were recorded by a digital electroencephalograph (Nihon-

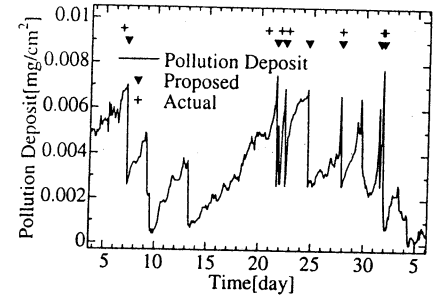


Figure 3: Result of automatic decision making by use of the processed data of Karatsu substation during the period of 3rd September to 5th October 1996.

### Spike-and-slow-wave

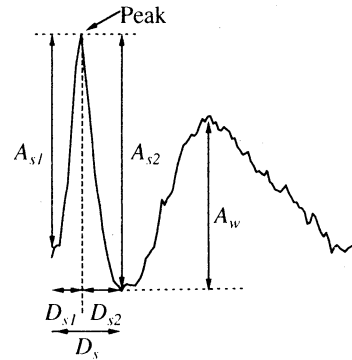


Figure 4: Template waveforms and parameters extracted from real EEG spikes and adopted for selecting candidates of spikes and spike-and-slow-wave.  $A_{s1}$  and  $A_{s2}$ ; amplitudes for spike,  $D_s$ ; duration of spike and  $A_w$ ; amplitude of slow wave.

Koden EEG2100) in Kyoto University Hospital. Nineteen electrodes ( $Fp_1, F_3, C_3, P_3, O_1, Fp_2, F_4, C_4, P_4, O_2, F_7, T_3, T_5, F_8, T_4, T_6, Fz, Cz$  and  $Pz$ ) were placed in accordance with the International 10-20 Electrode Placement System[11]. Thirty awaked EEG time series of bipolar montage were subjected for the automatic spike detection. The EEGs were sampled at an interval of 5 msec with a time constant of 0.3 sec and high-cut frequency of 60 Hz. Data were acquired from

10 patients with epilepsy of various forms, aged 29-69 years, and a total of 355 sec long EEGs was analyzed.

For establishing the automatic spike detection based on the artificial realization of on-off human decision making procedure described in Chapter 2, 115 spike waveforms were selected from the 30 bipolar time series of the topographical portions at the 75 spikes detected by visual inspection, and 22500 activities other than spikes (non-spike activities) such as alpha wave, EMG artifacts and so on were selected from the recorded EEG (355 sec  $\times$  30 derivations).

**Parameter extraction** The non conditional probability for Go (spike) and that for Nogo (non-spike) were

$$P(1) = 115/(115+22500), P(0) = 22500/(115+22500). \quad (12)$$

Parameters for detecting spike candidates were extracted from the EEG records, as seen in Fig. 4. A total of 6 parameters (see Figure 4); sharpness of spike ( $A_{S1}/D_{S1}$  and  $A_{S2}/D_{S2}$ ), its duration ( $D_S$ ), amplitude of slow wave ( $A_W$ ), prominence of slow wave from background activities ( $A_W/A_T$ ) and waveform of Sp-SW ( $R_{SW}$ ), were calculated. Notation  $A_T$  ( $= (4\sqrt{S_\delta} + 4\sqrt{S_\theta} + 4\sqrt{S_\alpha} + 4\sqrt{S_\beta})/4$ ) indicated a mean amplitude of whole background activities in delta (0.5-4 Hz), theta (4-8 Hz), alpha (8-13 Hz) and beta (13-25 Hz) bands, and  $R_{SW}$  was a cross-correlation coefficient.

Figure 5 shows histograms of six parameters for selecting candidates of spike calculated from 115 samples of spike and 22500 non-spike activities. Although parameter distribution of the two groups were nearly superimposed for duration ( $D_S$ ), other five parameter distributions concerning the sharpness ( $A_{S1}/D_{S1}$  and  $A_{S2}/D_{S2}$ ), amplitude of slow wave ( $A_W$ ), prominence of slow wave from background activities ( $A_W/A_T$ ) and waveform ( $R_{SW}$ ), especially that for waveform ( $R_{SW}$ ), were distinguishable.

**Spike candidate selection based on conditional probability** Conditional probability (eq.(7)) for detecting candidates of spike were calculated by using multi-dimensional normal distribution. Equation for detecting spike candidates was adopted and any wave that satisfied the above condition was selected as a spike candidate.

**Final judgment of spike** Judgment of spike was done by taking into account the topographical distribution of spike candidates selected, for each left and right hemisphere independently.

If the total number of derivations which contain the peak of selected spike candidate within optional 40 msec in the left or right hemisphere was greater than or equal to four, then those candidates in the short period of 40 msec were finally judged to be a spike.

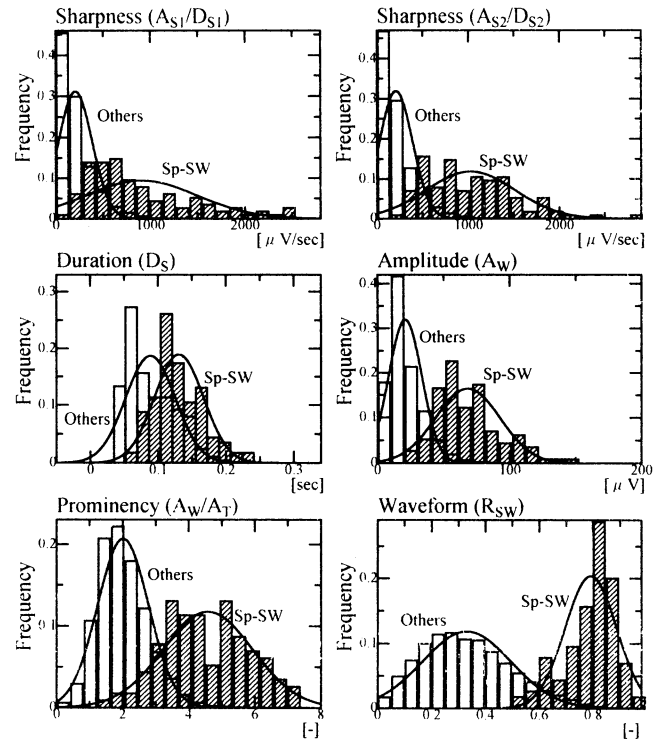


Figure 5: Normalized histograms and their probability density functions for discriminant parameters of spike. Shaded bars represent histograms of parameters for spikes, and open bars indicate those for non-spike activities. Solid lines show the probability density functions of normal distribution for respective periodograms.

### 4.3 Evaluation of the automatic spike detection

Threshold value  $\lambda$  in spike candidate selection procedure was set to the three different values (0.01, 0.5, 1.0), and the automatic detection of spikes for respective thresholds was applied to all data.

Fig. 6 shows an example of the results obtained by the automatic spike detection. By using  $\lambda$  of 1.0, only the definite spikes were detected. With  $\lambda$  of 0.5, two spikes were detected in total, and with  $\lambda$  of 0.01, all spikes were correctly detected, but some false positive spikes occurred.

The results of the automatic spike detection for all the data are discussed. Evaluation of the automatic spike detection was based on comparison with the results of the visual inspection of EEG data by an EEGer. Seventy-five spikes were detected by an EEGer from the total 355 sec time series obtained from 10 subjects. Twenty-seven spikes out of 75 were detected and 9 false positives occurred with  $\lambda$  of 1.0. Sixty-five spikes out of 75 were detected and 10 false

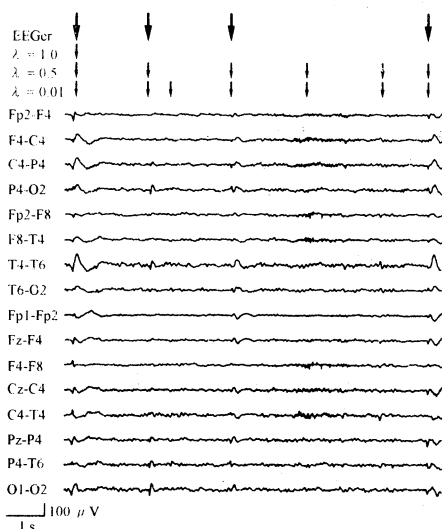


Figure 6: Example of the automatic spike detection by adopting 3 different thresholds  $\lambda$  ( $= 1.0, 0.5, 0.01$ ). 10 sec record of 16 EEG time series of bipolar derivation over the right hemisphere are displayed, and the detected spikes for respective threshold values  $\lambda$  are indicated by arrows on the top of the figure. Visual inspection of spikes for recorded EEGs done by an EEGer is also indicated by arrows on the top.

positives occurred in the condition of  $\lambda$  of 0.5. With  $\lambda$  of 0.01, all the spikes detected by an EEGer were detected correctly, but the number of false positives was 368 in total.

In this study, three levels of threshold values 0.01, 0.5 and 1.0 were adopted. However, the values of thresholds can be selected freely in accordance with the respective purpose of the spike detection of respective occasions. Therefore, the proposed method will be an effective tool for clinical use for many purpose.

## 5 Conclusion

A schematic procedure for artificial realization of on-off human decision making was derived based on conditional probability by use of acquired actual data. The proposed method was successfully applied to the problems of insulator washing timing and the spike detection on EEG record. By through the successful results for actual decision problem, the proposed decision making method based on conditional probability of acquired database was proved to be usable in wide field of realization of artificial human decision problems. The proposed method based on the conditional probability of mixture of discrete and continuous events is closely related to likelihood test method[5], fuzzy logic[10], mutual entropy[8] and voice decision[4], and is understood as an extended concepts of those related topics.

## Acknowledgments

The authors are grateful to Prof. H. Shibasaki (Medical School of Kyoto University) and Mr. T. Taniguchi (Kyushu Electric Power Company) for their valuable discussions on human decision making in their specific fields.

## References

- [1] Electric Joint Research (1979), vol. 35, no. 3 (in Japanese).
- [2] H. Shibasaki (1974), "Attempt toward the Standardized Interpretation of clinical EEG", *Clinical Electroencephalography*, vol. 16, No. 5, (in Japanese).
- [3] S. Goto, M. Nakamura, N. Nanayakkara and T. Taniguchi (1995), "Reliable Automatic Decision Making for Washing the Polluted Insulators in Coastal Substations", *Conference on Electrical Insulation and Dielectric Phenomena*, pp. 408-411
- [4] H. Kobatake (1987), "Optimization of Voice/Unvoiced Decisions in Nonstationary Noise Environments", *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 35, no. 1, pp. 9-18
- [5] S. Kullback (1978), "Information Theory and Statistics", Peter Smith, USA
- [6] M. J. Lighthill (1959), "An Introduction to Fourier Analysis and Generalized Function". Cambridge University Press, New York
- [7] A. Papoulis (1965), "Probability, Random Variables and Stochastic Processes", McGraw-Hill Kogakusha, Tokyo
- [8] C. E. Shannon (1949), "The Mathematical Theory of Communication", The University of Illinois Press
- [9] T. Sugi, M. Nakamura, A. Ikeda and H. Shibasaki (1997), "Automatic EEG Spike Detection by Use of Adaptive Decision Criteria to Individual EEG Records", *11th IFAC Symposium on System Identification (SYSID'97)*, CS20-4, pp. 1313-1318
- [10] L. A. Zadeh (1973), "Outline of a New Approach to the Analysis of Complex Systems and Decision Process", *IEEE Trans. Sys. Man & Cybern.*, vol. 3, no. 1, pp. 28-44
- [11] H. H. Jasper (1958), "Report of the Committee on Methods of Clinical Examination in Electroencephalography", *Electroenceph. clin. Neurophysiol.*, 10, pp. 370-375

# Artificial life with play instinct

Shinichi Tamura

Division of Functional Diagnostic Imaging, Osaka University Medical School  
D11, 2-2 Yamadaoka, Suita City, Osaka 565-0871, JAPAN  
Tel: +81-6-879-3560 Fax: +81-6-879-3569 tamuras@image.med.osaka-u.ac.jp

Shoji Inabayashi Youji Katou  
System Sogo Kaihatsu Co. Ltd.

## Abstract:

In this paper, we propose a model to generate a group of artificial lives capable of coping with various environment which is a set of tasks, and like to show the plays or hobbies are necessary for the group of individuals to maintain the coping capability with various changes of the environment as a whole. This may be an another side of that the wide variety of the abilities in the group is necessary, and if the variety in a species decreased its species will be extinguished.

Keywords: Artificial life, Artificial mind, Play, Genetic algorithm, Ability, Human resource

## 1. Introduction

Life is a highly rationalized organization. There often exists hidden rationality even in an irrational action at a glance. It is often said that in the organization the top a-third people work hard and draw the organization, but the bottom a-third are lazy and work as brakes for the development of the organization. However, if a new organization is made with only this bottom a-third, also new top a-third people are born from them as excellent constituents and draw the organization. Generally speaking, each individual has their own peculiar and unique abilities, and he or she exhibits the hidden ability when it is requested. Human society is composed of divided works and specialized individuals with various explicit and implicit abilities.

The play is often referred as an opposite action to the diligence. However, like the play at housekeeping of the children, if we consider it as the training occasions of the social roles, the meaning of the play becomes largely different. The human society is requested to continuously cope with variously changing environment with their variety of abilities. At ordinary times most part of such abilities are hidden as it is, and a part of it is revealed as the play and trained.

In this paper, using genetic algorithm we generate a group of artificial lives capable of coping with various environment which is a set of requested tasks. Under such situation, we like to show that the plays or hobbies are automatically generated as a necessary instinct or they are a natural result of revolution of the group of individuals in order to maintain the coping capability with various changes of the environment as a whole. This may be an another side of that the wide variety of the abilities in the group is necessary and if the variety in a species decreased its species will be extinguished.

This is an attempt to generate an artificial mind which has two aspects of "diligent to his job" and "apt to run to pleasure." We like to show the latter does not have a negative meaning but also has a positive meaning to cope with variously changing environment as a group.

Though unfortunately at this point of time, the simulation has not finished yet, outline of the setting of the world around artificial lives with mind is shown in the following:

## 2. World model

### 2.1 A set of all kinds of human abilities

We assume that human abilities in this paper are countable such as  $a_1$ =[swift-of-foot],  $a_2$ =[patient],  $a_3$ =[skillful-in-calculation],  $a_4$ =[fair-spoken],  $a_5$ =[rich-in-leadership],  $a_6$ =[rich-in-curiosity],  $a_7$ =[rich-in-competitive-spirit],  $a_8$ =[aggressive],  $a_9$ =[fond-of-fishing], etc. The set of such abilities is defined as

$$A = \{a_1, a_2, \dots, a_n\}. \quad (\text{Ability set}) \quad (1)$$

Null element  $\phi$  may be included in A to make the following expressions simple, by which the sizes of some of the following sets may be made fixed.

## 2.2 Environment

An environment  $E_t$  at time  $t$  is expressed as a set of requested tasks in the environment such as,  $T_{old-times\ 1} = [\text{protect-from-wild-animals}]$ ,  $T_{recent-ages\ 1} = [\text{publish-book}]$ ,  $T_{recent-ages\ 2} = [\text{open-a-store}]$ ,  $T_{recent-ages\ 3} = [\text{negotiate-with-other-company}]$ ,  $T_{today\ 1} = [\text{set-computer-business-up}]$ , etc., which is expressed as:

$$E_t = \{T_{t1}, \dots, T_{tnt}\} ; \quad t=1,2,\dots \quad (\text{Environment}) \quad (2)$$

The environment changes time to time.

## 2.3 Task

Each task  $T_i$  is expressed by a set of requested abilities to accomplish it as:

$$T_i = \{a_{i1}, a_{i2}, \dots, a_{in_i}\} ; \quad i=1, \dots, N_T. \quad (\text{Task}) \quad (3)$$

For example,

$$T_1 (= \text{task of } [\text{protect-from-wild-animals}]) = \{\text{quick-motion, cool-mind, good ear}, \dots\}.$$

$$T_2 (= \text{task of } [\text{political-action}]) = \{\text{fair-spoken, planning-ability}, \dots\}$$

## 2.4 Play

On the other hand, there are a set  $P_t$  of all hobbies including plays, self-development, or volunteer activities, etc. to which people begin or join voluntarily at time  $t$ . That is

$$P_t = \{R_{t1}, \dots, R_{tnt}\} ; \quad t=1,2,\dots \quad (\text{Play set}) \quad (4)$$

For example,

$$P_t = \{\text{soccer, go-game, marathon, reading, travel, fishing, volunteer-activity}, \dots\}$$

Each play also requests set of abilities:

$$R_i = \{a_{i1}, a_{i2}, \dots, a_{im_i}\} ; \quad i=1, \dots, N_R. \quad (\text{Play}) \quad (5)$$

For example,

$$R_1 (= \text{play of } [\text{fishing}]) = \{\text{interest-in-fishing, driving car, patient}, \dots\}$$

$$R_2 (= \text{play of } [\text{go-game}]) = \{\text{interest-in-go-game, pattern recognition, spatial reasoning, tactical sense}, \dots\}$$

Abilities of [interest-in-\*] is always necessary to begin the play of \*. The set  $P_t$  of plays also changes time to time according to the fashion. Only the difference between the task and the play is that the task is passively given or controlled by the environment, and the play is sought voluntarily or begun by urge and hard to stop.

## 2.5 Individual

Each individual  $I_k$  has his abilities and their skills:

$$I_k = \{a_{k1}, \dots, a_{kn_i} ; s_{k1}, \dots, s_{kn_i} ; p_k\} ; \quad k=1,2,\dots,N_I. \quad (\text{Individual}) \quad (6)$$

where  $s$  is a corresponding skill and  $p$  is a rate of spending time for the play where the hidden as well as the explicit abilities are trained. The set of individuals is generated and evolved by GA algorithm. Skills are raised by the experience in the job task or in the play but not inherited. The abilities and the ratio  $p$  are inherited across the generations. When one has more than two hobbies, time ratio  $p$  should be divided among them.

## 3. Formulation

A simple example of the world model is shown in Fig.1. The outside environment is always changing. It is the origin of the evolution. In this chapter, some basic formulas of the simulation are given.

### 3.1 Skill

We assume here the skill is given by

$$s_{ki} = 0.5 + \sum_t [\alpha_T \times p_k + \alpha_R \times (1-p_k)] \quad (\text{Skill}) \quad (7)$$

which is roughly proportional to the experienced time (years, or unit years) the ability  $a_{ki}$  was used in the task or the play.  $p_k$  and  $1-p_k$  are rates of time used for the play and the task, respectively. 0.5 is an initial skill when the individual has no experience.  $\alpha_T$  takes 1 when the ability  $a_{ki}$  is used in the task, and 0 when not used.  $\alpha_R$  is that of the play. The value of skill will be upper limited by their life-span automatically.

### 3.2 Assignment

When the environment  $E_t$  is given, an individual most fitted to the task (refer 3.3) among randomly selected unassigned  $N_e$  (e.g., five) individuals is assigned to the task. Individuals not assigned to any task is called "window-side-folks" and direct his energy to the play with the time rate of

$$\min[1, b \times p_k]; \quad b \geq 1 \text{ (e.g., } b=2 \text{)}. \quad (8)$$

Then the skills used implicitly as well as explicitly in the play will be increased as much.

### 3.3 Fitness to task

The fitness of the individual  $I_k = \{a_{k1}, \dots, a_{kn_i}; S_{k1}, \dots, S_{kn_i}; p_k\}$  to the task  $T_i = \{a_{i1}, a_{i2}, \dots, a_{in_i}\}$  is given by sum of the skills used in the task as

$$f_k = (1-p_k) \times \sum_{j=1}^{n_i} S_{kj} \quad \text{such that } a_{ij} = a_{kl} \text{ for some } l \text{ in } \{1, 2, \dots, n_i\}. \quad (9)$$

(Fitness)

If there are two same abilities in an individual, its skill is counted twice, and so on. That is, for example, if one has three genes of the same ability of [swift-of-foot], he may have three times of the skill of the [swift-of-foot]. On the other hand, if he has no such ability,  $S_{ki}$  is counted as zero. Though this process may be more nonlinear, for the ease of modeling it made linear.

### 3.4 Fitness to play

Also for the plays we can consider the same fitness, and each individual selects the most fitted  $R_i$  among the set of hobbies specified by ability of [interest-in-\*] and practice it. If one has no such ability, he will spend the leisure time with rate  $p$  idly; i.e., his abilities are not trained in this time.

### 3.5 Performance of whole world

Performance of the whole world is given by

$$Q = \sum_{i=1}^{N_i} f_i^* \quad \text{(Performance)} \quad (10)$$

where  $f_i^*$  is  $f_k$  of an individual  $I_k$  who is practically assigned to the task  $T_i$ . Note that hobbies are not evaluated at all in this model.

### 3.6 Evolution of individuals

Genetic algorithm is employed to make the evolution of the group of the artificial lives (set of individuals). At every year (or unit year), some percentages of the individuals with fitness in order from the lowest or randomly to some extent are selected and erased. Then, those with the highest fitness are multiplied.

### 3.7 Geno-type

In order to simulate the artificial lives we must map the model to geno-type. It may be most simple to represent the individual  $(\{a_{k1}, \dots, a_{kn_i}; p_k\})$  by a sequence of binary numbers.

The environment is given from the outside to the group of individuals. It requests the tasks. On the other hand, the hobbies are generated as a result of the evolved genes. To do so, some specific binary sequences on the gene should be interpreted as [interest-in-\*] with which he or she is represented as being fond of that kind of play and begin to do it. The set of trained abilities in each play is given also from the outside.

## 4. Expected result of simulation

It is expected that a group of individuals with wide variety of abilities as a whole is generated by the effect of varying environment. In other words, required wide variety of abilities to cope with variously changing environment are embedded distributively in the group. As a result each individual becomes to have his own personality and uniqueness. Individuals constitute the world (group) by sharing tasks required in the present environment. However, the whole abilities embedded in the group distributively are not only ones required in the present environment but also in its possible changes. Some part of the abilities embedded into an individual will appear as plays or hobbies. It will be expected that in spite of only evaluating the fitness of abilities of the individual to the changing environment, not only the variety of the abilities for the tasks but also the variety of the abilities for plays will be appeared. As a result, three types of individuals will be generated. The first type of the





# Autocatalysis as Internal Measurement and Origin of Programs

Shin'ichi Toyoda

Department of Computer and Systems Engineering,  
Faculty of Engineering, Kobe University, Nada, Kobe,  
Japan

## Abstract

A biological system is regarded as a complex large autocatalytic network consisting of many chemical reactions, and it is regarded as a kind of machine which is controlled by DNA regarded as a kind of program which has a all information of a system inherited from original to copies. In this session, I proposed a model for a auto-catalysis network with internal measurement theory which is theory for constructive understanding of situation objectively inseparable to measurement object and an observer, constructing as a interface between observed finite tree-structured rule of a system and non-observed infinite loop structured rule. In simulations of this model, we see various types of network. these are classified into some classes by its size and structure. In one of such networks, there is a network regarded as a system which has a type of element or types of element regarded as a program. We can regard this system as a program and a machine controlled by a program. I argue that this classification of a network seen in the model is regarded as hierarchy of a network. I argue that evolution of network and origin of program is able to be understood by a problem of measurement process.

## 1 Introduction

Recently, research for real biological system has been much advanced. For many phenomena, biochemical or biophysical processes have been revealed. We now have much concepts of biological processes, and can apply them to an artificial system. Generally, a most characteristic and important property of biological system is regarded as self-replication and auto-catalysis. They depend on each other and construct the life. They realize by very complicated process. Then in (real) biological system, it is a considerable problem how such a complicated systems were generated in primitive earth envi-

ronment. There are some approaches to this problem. In any approach, the aim is mainly to realize spontaneous organization of complex functional system from chaotic environment. In theoretical researches of complex systems, it is often explained by multi agent system showing an edge-of-chaos property, or self organized criticality (SOC). These model represent as a map of real existing biological systems.

On the other hand, however, there are some assignment about paradoxical situation arisen by mapping from real system to formal system[1]. A model as map is constructed by covering the paradoxical situation implicitly. There is essential difficulty when constructing a model of real system. Some authors argued that this paradoxical situation drives a dynamics of the life and other natural systems[2, 3], and this process is called INTERNAL MEASUREMENT.

In this paper, I argue about an autocatalysis driven by the internal measurement, and about emergence of a program-machine structure in a system.

## 2 Internal Measurement and Biological System

A biological system consists of many chemical reactions and many physical processes. They are united highly functionally, thus maintain a life. On the other hand, a biological system can be recognized as a hierarchically associated network. we can find a formal description of process of life without its contents. A form that a life is controlled by a gene as a program is most essential one of a formal view of life. There are many researches in this program-machine view of life (for example [5]). They separate a program and a machine a priori. It is available for taking the evolution of the biological system, however, ineffective for the emergence of a biological system itself. In microscopic view, there is no difference of status between a program and a machine. Therefore, a problem of the emergence of life is not the emergence of DNA, and other many organic substance as a reactant, but the emergence of a specific status from complicated chemical reaction.

The status of program only comes from a view point to be able to overlook a whole system. If the system

has the structure and use it, the system must also recognize the whole of itself and find a rule in it. It arises a lot of difficulty. Because, we can not recognize any system in infinite precision. What we, and a system itself, can do is to observe a system in finite, and indeterminate, precision, finite times, and finitely long. Then we construct a structure of system. That is, this structure of system is, so to say, virtually. But, it works in real world. Then, a problem focuses a system's interface between finite measurement and infinite structure in itself.

The concept that interface between two different logical status drives a evolution has been argued by some authors[6, 7]. It is the concept called INTERNAL MEASUREMENT. In paper [7], Gunji et al. present a model for interface between finite and infinite states, called HYPER-DILATION, introducing a concept called Measurement-Oriented Physics. In this model, a interface between infinite and finite states is realize as transformation between tree-structured and loop-structured rule.

I apply this tree-loop interface to construct a model for autocatalysis. The essence of process of my model are, 1)observing the system and making a tree structure rule satisfying the observation, 2)transforming this rule to loop form and applying this rule to next transition of the system. It may seems simple, but the concretely formalized model is too complicated to represent completely in this short paper. It may be impossible to re-construct the model by reader, however, I would attempt to represent essence of this model as strict as possible.

This is an automata fashioned model representing a abstract chemical reaction model, one substrate-one product reaction catalyzed by one catalyst, that the simplest chemical reaction. Each substrate is regarded as a particular state of the agent sight. Namely, the sights regarded to be in different state are regarded as different substrate.

The system consists of  $M$  agents that are represented by binary decimal sequence such as  $a_i = 011001110101 \dots$ . The value of the  $k$ th precision of the  $i$  site is denoted by  $a_i^k \in \{0,1\}$ . This sequence is  $N$  length.

The transition rule of an agents in the system consists of the rules applied to each binary triplet, i.e.  $f_k : \Omega^3 \rightarrow \Omega$ . Where  $k$  is a number of the agent, and  $f_k$  is related to the state of the agent, then one of the arguments is the value of focused precision of the agent itself. One of the arguments of  $f_k$  represents a rule of the lower level, i.e. it is just the value of one lower level. the last argument is the value of a agent coupled with the concerned site, which is randomly chosen. Then the rule of the system

$F$  on the site  $i$  is expressed as

$$F_i(a_j) = f_{i,0}(a_j^0, f_{i,1}(a_j^1, f_{i,2}(\dots))) \quad (1)$$

In observing turn, this function sequence is constructed. For infinitely long sequence of  $a_i^k$ , it would need a infinite sequence of  $f_k$ . However, an observed system is finitely large then there exist a finite sequence of function satisfying transition already occurred. It would be a tree-structure which each branch has different depth (Fig. 1). Each terminal of branch is identified by the transition rule, so it is reasonable to regard it as one species of element.

In application turn, this finite tree rule must be applied to virtually infinite sequence. Therefore, tree rule is converted to loop structure rule, as shown in Fig. 1. Then, the loop rule is applied to each randomly chosen pair of agents. Through this two turn, the system progressing by one time step. Subsequently, back to observing turn, the process is repeated.

### 3 Simulations

I simulated this model by computer. A number of agents,  $M = 10000$ , and  $N = 30$ . In simulation, transformation from one agent to another was seen. Species of agent was determined by terminal of branch of tree rule. A form of tree changes at each time, so a predecessor was adopted.

There were some class of networks in simulated system. Some examples are shown in figure. In figure, a square represent a species of agent, and a number in square represent species number. An arrow represents a transformation, and a number besides an arrow represents a catalyst. The most simple class is identity transformation (Fig.2).

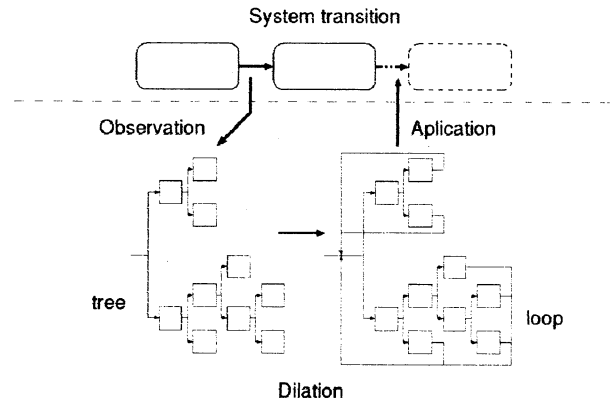
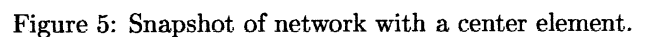
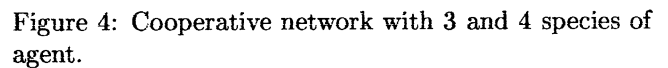
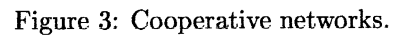
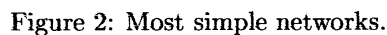


Figure 1: Conceptual image of process of hyper-dilation model

A rule of the system changes at each time step, so any network is not stable. Some times it becomes more complex drastic, and sometime becomes simpler one. It is not expectable at all.

In the proposed model, agents embedded in each rule are determined through measurement, though it is generally assumed that the reaction unit exists objectively without observer. In this modeling, a system can encompass the existence of an observer.



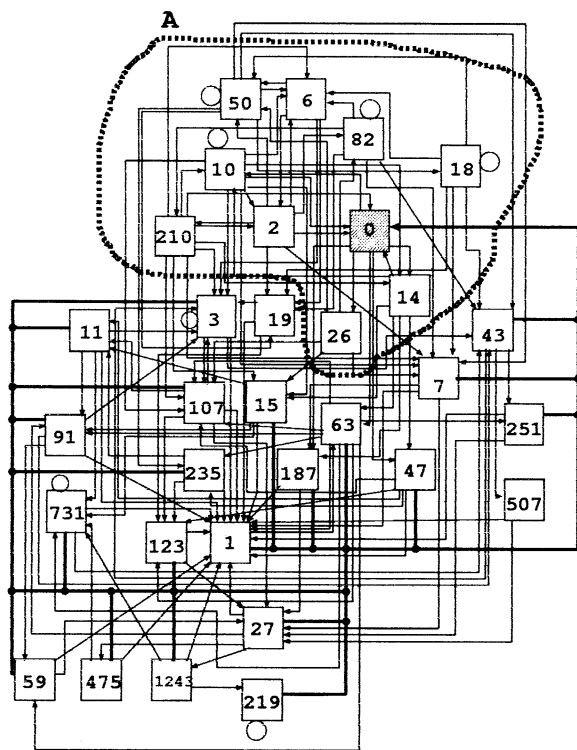


Figure 6: Complex network with a center sub-network.

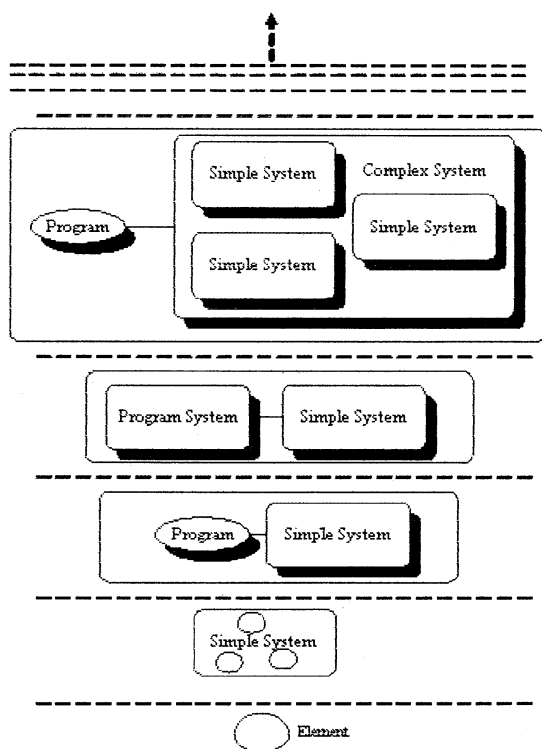


Figure 7: Image of hierarchy of networks in the model system

the Emergence and development of various auto-catalytic network and hierarchy of such networks are shown as advantageous points of this model. A network in which a reaction unit regarded as a program is included in these networks. A program produces and controls a network. This model consists of  $M$  elements, and each element is  $N$  length bit sequence. So, each element can be distinguished in  $2^N$  states. In this level, transition of an element is quite complex, or seems to be immethodical transition, because it is difficult to determine the transition rule, for a state space is larger than elements. Simple structures with a program emerges in some rough level of precision through measurement. This model can express that some structures degenerate from chaotic dynamics. Degeneracy of a network which is auto-catalytic of which has a program-like element suggests not that a specific structure of universal interaction between the objective elements in a system gives rise to a structural auto-catalytic network or program, but that a specific network or a program is degenerated through measurement process.

## References

- [1] H. H. Pattee, *Biosystem* 23(1989)281.
- [2] K. Matsuno, *Protobiology: Physical Basis of Biology* (CRC Press, Boca Raton, FL, 1989).
- [3] P-Y. Gunji, *Appl. Math. Comput.* 61(1994)231.
- [4] J. V. Neumann, *Theory of Self-reproducing Automata*, (ed. A. W. Burks, Univ. Illinois Press, 1966)
- [5] Hashimoto and Ikegami, *Artificial Life V*, (ed. C. G. Langton, K. Shimohara, The MIT Press, 1996)
- [6] P-Y. Gunji, H. Sadaoka and K. Ito, *Biosystem* 35(1995)213.
- [7] P-Y. Gunji, S. Toyoda, *Physica D* 101(1996)27.

# An immuno system model for generation of self tolerance and memory

Yoshihiro Ochi, Yoshiki Kashimori, and Takeshi Kambara

Department of Applied Physics and Chemistry,  
The University of Electro-communications, Chofu, Tokyo, 182 Japan  
e-mail: kashi@nerve.pc.uec.ac.jp

Six key words: immuno system; cellular automaton;  
chronicity; cytokine; memory state; tolerance.

## Abstract

To explain systematically the two quite different response behaviors, immunological tolerance and memory of immuno system, we proposed a model of immuno system in which we introduce a quantity "chronicity" which represents quantitatively the frequency of interaction of each immuno cell with antigens. When the magnitude of chronicity of an immuno cell is too high or too low, the cell makes no response to the antigens. The cell may attack antigen, only when the magnitude of its chronicity is within a certain range. The activity of the system is also controlled by two kinds of cytokines, positive cytokine which activates immuno cell, and negative cytokine which inhibits the activity of immuno cell. The model provides a unified view of the mechanism by which the tolerance and memory are generated.

## 1 Introduction

Normal immuno system responds to harmful (foreign) antigens, while the system becomes tolerant toward harmless (self) antigens, such as normal cells and tissues of host. The immuno system is able to judge what antigen the system should attack. Immuno system has also the function of memory. Once the system has responded successfully to certain antigens, the system memorizes the antigens. When the same antigen invades into host, the immuno system attacks much more quickly and effectively the antigen and prevents host from the reinfection. The tolerance and memory are the most important function that immuno system has. Therefore, a number of studies have been made to clarify the mechanism generating the tolerance and memory, and many hypotheses<sup>1-4</sup> have been proposed about the mechanism. However, there exists no model in which both the tolerance and memory may be systematically generated based on a unified mechanism.

In the present study, we proposed a model of immuno system which can explain systematically the generation of both tolerance and memory. We consider that an essential difference between self and non-self antigens arise from the difference between their amount in their host body. In our model for the unified mechanism generating tolerance and memory, we introduced a state variable of immuno cell, "chronicity" which represents quantitatively the frequency of interaction of each immuno cell with antigens. The experimental results that immuno cell response alters depending on the frequency of interaction of each immuno cell with antigens have been demonstrated<sup>5,6</sup>.

The difference between tolerance and memory may be explained based on the chronicity, as follows. The immuno cell specific to a certain self antigen interacts with the self antigens in a high frequency as shown in Fig. 1a, because the number of the self antigens in a host is enormous and they are supplied constantly. Then, the chronicity of immuno cells specific to the self antigens is always very high, and the immuno cells becomes inactive because of their high chronicity. On the other hands, the chronicity of immuno cells specific to foreign antigens is lower, because the number of harmful foreign antigens is less than that of self antigens, as shown in Fig. 1b. When the immuno cell has a middle chronicity, it can respond actively to foreign antigens, and stays in a memory state after the decrease of foreign antigens.

The model includes also two kinds of cytokines, positive cytokine which activates immuno response, and negative cytokine which inhibits the activity of immuno cell. Each immuno cell secretes one of the two kinds of cytokines depending on its chronicity; for middle chronicity, the positive cytokine is secreted, and the negative cytokine is secreted for high chronicity. Cytokine can diffuse in a host body. The generation of tolerance and memory depends not only on the frequency of interaction of the immuno cell with antigen but also on the distribution of cytokine in a host body.

We show here based on our computer simulation that the model may explain systematically the mechanism generating both tolerance and memory states by introducing a unique quantity, chronicity. We also show that the memory state has a property of self-organized criticality. We discuss the experimental plausibility of the role of chronicity.

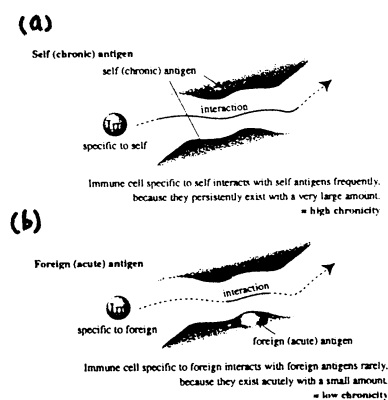


Fig. 1 Chronicity of an immuno cell specific to (a) self antigen and (b) foreign antigen. Chronicity of a cell represents a frequency of interaction of cell with the relevant antigens.

## 2 Model

### Interaction field

Antigens intruded into the host body are transported to a lymph node by lymphocytes and lymph fluid, and then they interact with immuno cells inside the lymph node. As shown in Fig. 2, we described the lymph node, which is main stage of immuno response process, as  $100 \times 100$  rectangular grid of unit spaces with a periodic boundary condition.

### Characters

We consider only six kinds of characters in this cellular automaton model; two kinds of immuno cells, two kinds of antigens, and two kinds of cytokines. In the natural immuno system, there are many kinds of immuno cells, such as B cells, helper T cell, killer T cells, and dendritic cells. Because our main purpose is to propose a reasonable model of mechanism generating systematically the tolerance and the memory, we consider artificial immuno cells whose function is obtained by unifying systematically the functions of those many kinds of immuno cells. We consider one kind of self-antigens and one kind of foreign antigens, and two kinds of artificial immuno cells responding specifically to those antigens. The system is composed of not only these immuno cells but also chemical compounds, cytokines. Although the natural immuno system has different kinds of cytokines, such as INF, TNF, and interleukin, we considered only two kinds of cytokines as the representatives of these cytokines; a positive cytokine which activates immuno cells, and a negative cytokine which inhibits the activity of immuno cells. Immuno cells interact specifically with antigens but nonspecifically with cytokines.

The physical state of each characters is represented by the status variables described in in Table 1. Specificity represents the fitness of receptors of the immuno cell with antigens. The cells attack more intensively the antigens as the specificity increases. Chronicity is a variable that represents effectively the frequency of interaction of immuno cell with antigens. As immuno cell interacts frequently with antigen, the chronicity increases, but as the interaction occurs more rarely, the chronicity decreases. The low threshold is a threshold of chronicity above which the immuno cell becomes active, while the high threshold is a threshold of chronicity above which the immuno cell becomes inactive as shown in Fig. 3.

### Creation and destruction of characters

Immuno cells were supplied from the external environment with a constant rate, and were proliferated in the active state generated by antigens, and eliminated by natural death. Antigens were injected from the outside, and proliferated with a fixed rate and were eliminated by natural death and by attack of immuno cells. Cytokine is secreted and absorbed by immuno cells and eliminated within a natural decay.

### Movement of characters

Immuno cells and antigens can move from  $(i, j)$  site to another site, which is chosen randomly among neighboring nine sites including  $(i, j)$ . The concentration of cytokine changes with diffusion, that is, depending on the difference of concentration among vertically and horizontally neighboring sites  $(i \pm 1, j)$  and  $(i, j \pm 1)$ .

### Proliferation of immuno cells and antigens

The probability of immuno cell replication is determined by

$$P_{pro} = \begin{cases} \frac{c_{ij}}{\theta_{ij}^H} & \text{for } 0 < c_{ij} < \theta_{ij}^H \\ \frac{1 - c_{ij}}{1 - \theta_{ij}^H} & \text{for } \theta_{ij}^H < c_{ij} < 1 \end{cases} \quad (1)$$

where  $c_{ij}$  and  $\theta_{ij}^H$  is the chronicity and the high threshold for the immuno cell at  $(i, j)$  site. The reproduced cells move immediately to several sites randomly chosen from nine neighboring sites. Immuno cells newly reproduced by replication have the same chronicity and the same high and low thresholds as those for the parent. However, the immuno cell inherit specificity is randomly chosen from three specificities,  $s_{ij}^{Im}$  and  $s_{ij \pm 1}^{Im}$ . Antigens go on replicating with a constant rate.

### Interaction between characters

Only when the immuno cells become active, they can kill the relevant antigens and proliferate. The probability  $P_{act}$  that the cell on  $(i, j)$  site is activated by antigens on the nearest neighboring sites is given by

$$P_{act} = \frac{1}{9} \sum_{i'=i-1}^{i+1} \sum_{j'=j-1}^{j+1} \exp[-(\frac{s_{ij}^{Im} - s_{i'j'}^{Ag}}{\sigma})^2], \quad (2)$$

where  $\sigma$  is the parameter determining the spatial range of interaction. When each immuno cell becomes active, it makes one of the three different actions depending on the values of its own chronicity  $c_{ij}$  and thresholds  $\theta_{ij}^L$  and  $\theta_{ij}^H$ .

(i)  $0 \leq c_{ij} \leq \theta_{ij}^L$

The immuno cell proliferates in the probability  $P_{pro}$  given by Eq. 1 and don't attack any antigen.

(ii)  $\theta_{ij}^L \leq c_{ij} \leq \theta_{ij}^H$

The immuno cell proliferates in the probability  $P_{pro}$  and kill an antigen on the same site in the probability  $P_{kill} = (c_{ij} - \theta_{ij}^L) / (\theta_{ij}^H - \theta_{ij}^L)$ . The immuno cell also secretes positive cytokine whose amount is proportional to  $P_{kill}$ .

(iii)  $\theta_{ij}^H \leq c_{ij} \leq 1$

The immuno cell proliferates in the probability  $P_{pro}$  and don't attack any antigen. The immuno cell secretes negative cytokine whose amount is proportional to  $(c_{ij} - \theta_{ij}^H) / (1 - \theta_{ij}^H)$ .

The chronicity  $c_{ij}$ , the high and low thresholds  $\theta_{ij}^H$  and  $\theta_{ij}^L$ , and the concentrations of positive and negative cytokine  $P_{ij}$  and  $N_{ij}$ , are updated by the difference equations with respect to  $c_{ij}$ ,  $\theta_{ij}^H$ ,  $\theta_{ij}^L$ ,  $P_{ij}$ , and  $N_{ij}$ .

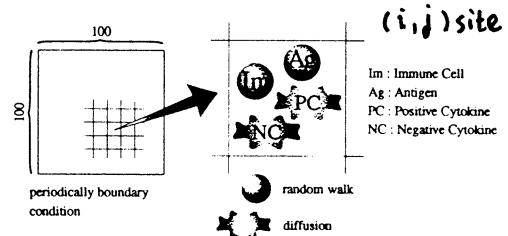


Fig. 2 Cellular automaton model of immuno system ( $100 \times 100$  rectangular grids of unit spaces). One immuno cell (Im), one antigen (Ag), one unit of positive cytokine (PC) and one unit of negative cytokine (NC) can be occupied in a unit space.

character	variables	expression
immune cell	specificity	$s_{i,j}^{im}$
	chronicity	$c_{i,j}$
	low threshold	$\theta_{i,j}^L$
	high threshold	$\theta_{i,j}^H$
antigen	specificity	$s_{i,j}^{ag}$
positive cytokine	concentration	$P_{i,j}$
negative cytokine	concentration	$N_{i,j}$

Table 1. State variables of four characters. (i, j) stands for ith row and jth column of  $100 \times 100$  grid.

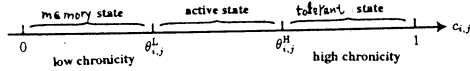


Fig. 3 Two kinds of thresholds of chronicity and three states of immuno cell classified by the thresholds.

### 3 Results

#### 3.1 Immuno response to foreign(acute) antigens

Figure 4 shows the response property of the immuno system to foreign antigens whose replication rate is low. In the initial state, the specificity of each immuno cell to the antigens is given randomly and all of the immuno cells are in the inactive states. At first a few foreign antigens are injected into the field. In the early stage of the simulation ( $t < 10$ ), the number of antigens increases because the injected antigen are replicated without interaction with immuno cells. At  $t$  around 10, the immuno cells whose specificity to the antigen injected is high start interacting with antigens. Then, the chronicity of the cells increases and as a result the value of low threshold decreases as shown in Fig. 4(b). Therefore, the probability that the cells kill the antigens increases and the cell secrete much positive cytokines. The number of activated immuno cells increases, because the virgin immuno cells are activated by positive cytokines. Thus, the antigens injected are decreased until the system stays in a steady state as shown in Fig. 4(a). After  $t=20$ , the state variables of four characters become constant. The steady state obtained corresponds to the memory state whose meanings is described in section 3-2.

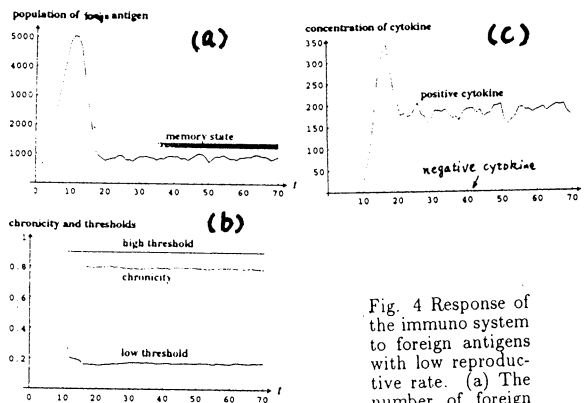


Fig. 4 Response of the immuno system to foreign antigens with low reproductive rate. (a) The number of foreign antigen. (b) Values of the chronicity, and the high and low thresholds. (c) concentrations of positive and negative cytokines.

#### 3.2 Memory state

We define memory state as equilibrium state where the number of antigen, the number of immuno cells, and the concentration of the positive and the negative cytokine are constant. Figure 5 shows a function of memory state to reinfection; the rate of decrease of antigen for the secondary injection of antigen is extremely faster than that for first injection of antigen.

Figure 6 shows the power spectrum for time variation of the number of antigen in memory state. The slop of  $P(\omega)$  is nearly -1. This indicates that the memory state is organized toward a critical state by a self-organized mechanism. That is, the memory state has both stable and unstable dynamics for small amount of injection of antigen.

#### 3.3 Immuno response to self(chronic) antigens

The self(chronic) antigens exist persistently in the field and their amount is very large. First, we inject some amount of virgin immuno cells into the field, where the value of specificity of each cell to the self antigens is given randomly. Because every cell contacts frequently with so many antigens, the chronicity of cells whose specificity is high increases rapidly. As a result, the population of the specific cells increases, but the self antigens are never decreased. Therefore, the value of chronicity of each cell exceeds the high threshold value. Then, the cell secrete negative cytokines and the secreted negative cytokines reduce the ability of immuno cells to attack the antigens. Thus, an immuno tolerance state occurs in the region where there exist dominantly the negative cytokines. In the region, there exist two kinds of temporal and spatial oscillations of cytokine concentrations with anti-phase; small amplitude oscillation for positive cytokine and large amplitude oscillation for negative cytokine. The small amount of positive cytokines play an essential role in maintaining the value of chronicity of each cell above the high threshold value.

In a case where the foreign antigens have a very high replication rate, the similar response process occurred and we obtained a stable tolerance state which is similar to the stable oscillation state for the self antigens.

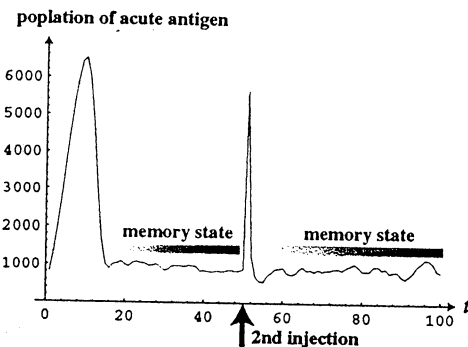


Fig. 5 Response of the immuno system in the memory state to a second injection of foreign antigens.

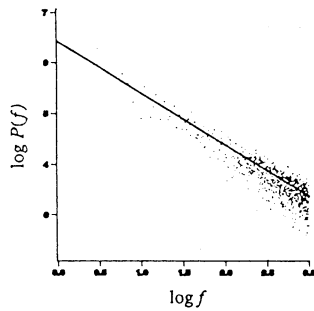


Fig. 6 Power spectrum of the fluctuation of antigen population in the memory state.

### 3.4 Immuno response to coexisting foreign and self antigens

We investigated the immuno response under the condition where there coexists the foreign antigens with low replication rate and self antigens. We started the simulation from the initial state where the cluster of self antigen is distributed within a small square area of  $100 \times 100$  grid. In an early stage, the foreign antigens were injected into the system. Figure 7 shows the response property. The immuno system suppresses only the self antigen, and keeps the number of foreign antigens being below a certain value. In outside of the self antigen area, the response property of the immuno system has the similar tendency to the property induced by foreign antigen alone. In the boundary region of the self antigen area, the positive cytokine increases intermittently as shown in Fig. 7(c). The intermittent increase prevents the foreign antigen from invading the self antigen area.

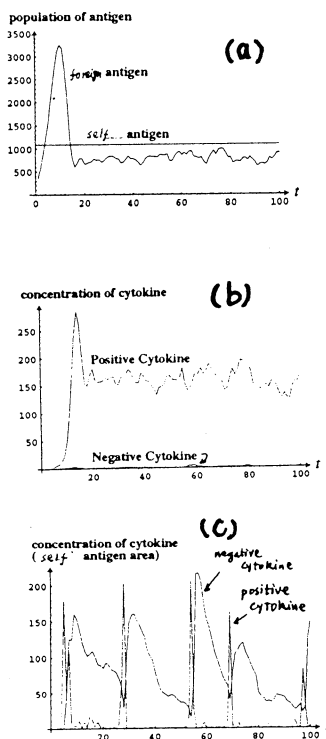


Fig. 7 Response of the immuno system in the case where foreign and self antigens coexist. (a) The number of antigens. Concentration of positive and negative cytokines in (b) outside of the self antigen area and (c) inside (boundary) of the self antigen area.

## 4 Conclusion

To understand systematically the immuno tolerance and memory states, we have introduced a variable "chronicity" which represents the frequency of interaction of immuno cell with antigens. When the immuno cell has a high chronicity, the immuno cell is inactive and the system stays in a tolerance state, while for a medium value of chronicity, the immuno cell is activated and can attack antigens. For a low value of chronicity, the system stays in the memory state. Then, cytokine is secreted depending on the activity of the immuno cell, and controls the immuno system by facilitating or suppressing the immuno activity.

Although the chronicity has not been yet directly observed, the fact that immuno cell response is changed depending on the frequency of interaction of immuno cell with antigen has been demonstrated experimentally<sup>2,5</sup>. These experiments suggest that the microscopic origin of the chronicity may come from dynamical properties of  $Ca^{2+}$  channels and the other channel proteins of immuno cell.

It is well known that T cells, which respond to self antigens during development, are eliminated in thymus. Also B cells, which strongly respond to self antigens during differentiation into B cell in the bone marrow, undergo negative selection. However, it is hard to consider that all immuno cells, which possibly induce autoimmunity, are eliminated during development. A part of those cells possibly can escape from the negative selection, and reach peripheral site of body. Although some theories<sup>6</sup> about the mechanism of acquisition of self tolerance have been proposed, there can hardly be seen clear relation among these theories. On the other hand, the present model provides a unified view of the mechanism of tolerance in both thymus and periphery.

## References

- [1] Zinkernagel RM(1996), Immunology through viruses, *Science* 271:173-178
- [2] Viola A and Lanzavecchia (1996), T cell activation determined by T cell receptor number and tunable threshold, *Science*, 273:104-106
- [3] Jerne NK(1974), Toward a network theory of the immuno system, *Ann. Immunol(inst. Pasteur)* 125C
- [4] Jerne NK, Idiotypic networks and other preconserved ideas, *Immunological Rev.* 79
- [5] Healy JI et al. (1997). Different unclear signals are activated by the B-cell receptor during positive versus negative signaling, *Immunity* 6:419-428
- [6] Matzinger P(1994), Tolerance, danger, and the extended family, *Annu. Rev.* 12:991-1045



# Simulations of Group Action of Artificial Honey Bees

Yasunori Niino

Johokagaku High School

Masanori Sugisaka

Department of Electrical and Electric Engineering , Oita University

700 Dannoharu, Oita City, Oita Prefecture, 870-1192 Japan

## 1 Introduction

The goal of the research of an artificial life seems to create a human-type robot. But, we think it will be achieved in the far future, because we human beings are created quite complicatedly and have the highest intelligence among the creatures.

On the other hand, there are many creatures like insects which have only instinct. At the first sight, they move fast and behave as if they have any intelligence or mind or will. But, in reality, they are only repeating monotonous actions observing simple rules. So we attempted to simulate the behavior of honey bees on defending their comb.

Though a honey bee is small and its poison is not so strong that it cannot kill an enemy by oneself, they can drive away or kill enemies by attacking in a group. Especially Japanese honey bees can kill hornets. They surround a hornet in a group, make a ball by themselves, make the temperature in the ball high, and kill the hornet.

So we think that it is useful to create insect-like robots and make their group. This paper shows that we can simulate the behavior of bees on defending their comb by using very simple rules.

## 2 Methods

We set an imaginary square field on a computer screen where honey bees live. The top-left corner is the comb, from where they fly away in all directions to collect honey, and to where they fly back after a certain time. They can fly in an original direction avoiding the other honey bees or obstacles.

The former is realized by the combination of a timer and the information on locations. The former is realized by searching, randomly for a certain times, a possible direction to move, if the original direction is obstructed.

If a honey bee find an enemy, it issues an alert to its colleagues and begin attacking the enemy at once. Other bees also begin moving in the direction and join the attacking at once.

Here, we assumed a case between Japanese honey bees and hornets. We set that the Japanese honey bees which attacked a hornet die one by one in the order of arrival, and that if the number of Japanese honey bees which surround a hornet reach to a certain value, the hornet die.

## 3 Results

We illustrate the results of the simulation from fig.1 to fig.3.

Fig.1 shows the state of honey bees which

are collecting honey. There, we can understand that many honey bees gather around their comb, and some other bees are out for collecting honey. The state of distribution of honey bees always change; therefore the number of bees which stay in the comb also change.

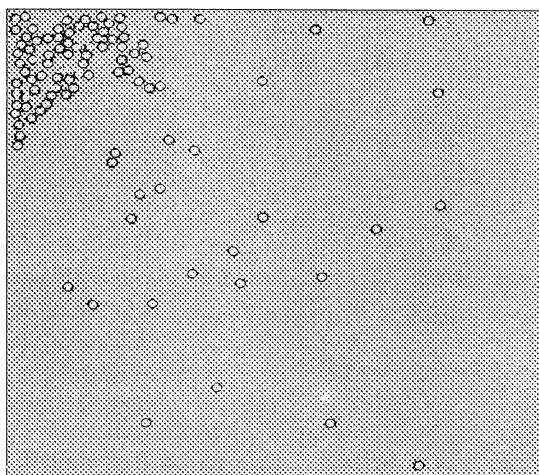


Fig.1

Although the rules we used in this program were very simple, we were able to simulate the natural action of bees to some extent.

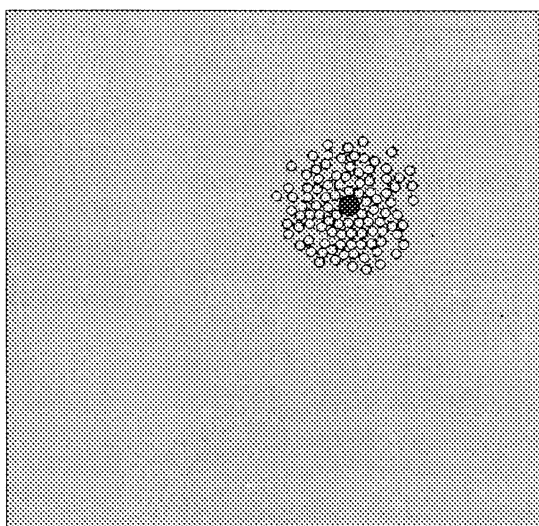


Fig.2

Fig.2 shows that Japanese honey bees are attacking a hornet. We can understand that honey bees gather around the hornet and surround it.

Fig.3 shows the instant of the death of a hornet. After the death of the hornet, honey bees are restored to the state of collecting honey as Fig.1.

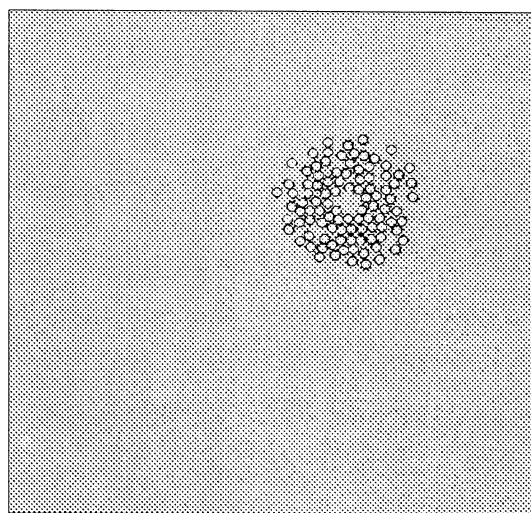


Fig.3

#### 4 Conclusion

As we see here, a lot of individuals which comply with simple rules can act as if they were under the command of a ruler. This fact implies that robots which do not have high intelligence like human beings also can do some intelligent, or intelligent like, jobs by forming a group.

In these days, the mass production technology of industry is surprising. We think that producing a group of low intelligent robots by using this technology is more possible in the near future than producing a human-type robot.

#### References

- [1] Katsura Hattori, "The World of Artificial Life", Ohm Sha, 1994
- [2] Takanori Shibata · Toshio Fukuda, "The Near Future of Artificial Life", Jijitsushin Sha, 1994

## Robust and Fast Color-Detecting using a Look-Up Table

Do-yoon Kim\*, Hyun-keun Park and Myung Jin Chung\*\*

*KAIST EE*

*\*nice@donghae.kaist.ac.kr and \*\*mjchung@ee.kaist.ac.kr*

**Abstract** - This paper describes a fast and simple approach to extract the desired color using a look-up table. Theoretically, the color distribution over an object is a scattering of points in RGB space lying somewhere along a ray from the origin toward the intrinsic color of the object. In practice, though, this is not the case due to object color variation, some specularities, camera noise, and the influence of nearby colors. We model this distribution by a look up table. The proposed approach has been successfully applied to the tracking of the position and orientation of robots with various lighting conditions.

The feasibility of the proposed approach is assessed by applying it to various cases.

**Key words** – robust, color-extracting, tracking, look-up table, MIROSOT

### 1. Introduction

In a robotic soccer game the current position and the direction of each robot are obtained from CCD cameras that overlook the complete field. The robots of each team are classified by team colors to be represented either by yellow or blue. Most teams use a color-based detection approach using an additional three colors because a single color on the robot is not enough to provide the directional information and the identification. However, with three different robots, it is not simple to find three distinguishable colors, as several others colors are assigned to various objects, such as green for the field, orange for the ball, and blue and yellow for the team colors. Furthermore the inevitable variations in lighting conditions over the entire area of the field increase the troubles in the vision system. It raises many problems. Though the same colors are used, they vary with position on the field. So the vision system doesn't identify each robot. These faults can bring about a wrong action by the robots. Many operators joining in a robotic soccer game are afflicted with the unbalanced lighting conditions.

So the operators need a vision system having robustness to various lighting conditions. Of course there are many solutions in the field of the color constancy to get the robustness to various illuminations, but they require a much higher computational cost<sup>6</sup>. They are not suitable for a real-time vision system, especially for robotic soccer requiring fast computation. The important factors in the vision system of robotic soccer are processing speed and robustness to variations in illumination. In the vision system, unfortunately, robustness and speed are often antagonistic.

The most natural way to represent the colors perceived by a vision system is by the (Red, Green, Blue) triplet, which is the output of the three types of light detectors. However this approach does not represent intrinsic colors in the best way, since the chromatic properties of the spectrum are not separated from the non-chromatic

ones, such as brightness in colors. A number of representations have been developed to differentiate the intensity within colors. Normalized RGB and HSI color models are good examples. But if we use these models we must also put up with decreasing the acquisition speed of the information to the positions of the robots, since these are obtained by a complex computation<sup>1,2</sup>.

Therefore it is necessary to find a robust but simple approach in RGB space. In this paper we focus on a robust and fast extracting approach in a desired color within the general PC-based environment, operated in Windows 9x with a relatively slow processor (150MHz Pentium). The proposed method doesn't require additional hardware like DSP.

The paper is organized as follows: Section 2 reviews the theoretic background about the behaviors of points represented in a given color in RGB space which lies somewhere along a ray. These tell us the drawback of the previous approach used by other researchers. In section 3 we propose a look-up table approach having two advantages of both speed as well as robustness in RGB space. The experimental results using real images show the proposed method is both fast as well as robust. We can easily track the position and orientation of three robots and their ball on a general PC at 60Hz. It is the maximum speed since the field's frequency of a NTSC signal is fixed at 60Hz. Section 4 concludes this paper.

### 2. Background

Most of the teams including our team have tracked the robots and their ball, dividing RGB space into two areas, which are the desired colors. In this case the dividing function takes the shape of a band pass filter with both minimum and maximum values. It is very simple and intuitive. This area takes the shape of the cube in RGB space like Fig. 1.

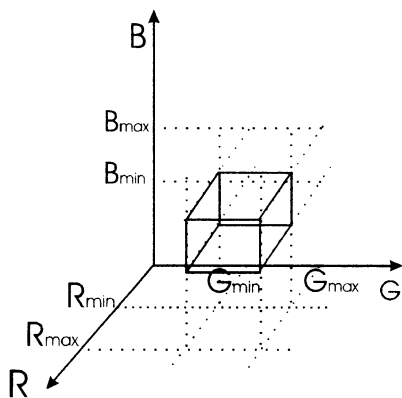


Figure 1. The shape of the passing area in RGB space

But, in order to track an object lying in various illuminations the bandwidth must be more widely adjusted. It oversees too much unnecessary areas. As a result, false detection occurs. Figure 2 shows this phenomenon. Figure 2-(a) is a real image that has various color samples on the field. The center field is brighter than the corners. The part of character 'Y' is a yellow color. Fig.2-(b) is the histogram of the yellow color on the center of the field and Fig.2-(c) is the histogram of the yellow color on the corners of field. Though these colors are really the same color, they have different distributions on the R, B and G axes. So when we set the min-max values with using Fig.2-(b) the outer patches are not as clear as shown in Fig.2-(d). When we set the values to use Fig.-(c), in this case, the center patches aren't searched as shown in Fig.2-(e). If we set the values with the union of Fig.2-(b) and Fig.2-(c) then many blobs appear in binary images after processing. This situation often occurs in robotic soccer games because spotlights beam into the playing ground to take a picture for TV broadcasting. In this condition

many operators can not detect yellow color patches. They demand to adjust the lighting conditions. These problems have caused that the changes in the lighting conditions lead to change color distribution with a corresponding drawback in the simple min-max values. Kliner<sup>5</sup> shows that a sample color distribution of an object with both Lambertian and specular reflectance components should be a tube-like cluster about the ideal ray, with branches or curvature toward the light source colors. Theoretically, we would expect that the color distribution over the object is a scattering of points in RGB space lying somewhere along a ray from the origin toward the intrinsic color of the object. In practice, though, this is not the case due to the object color variation, some specularities, camera noise, and the influence of near other colors as shown in Figure 3.

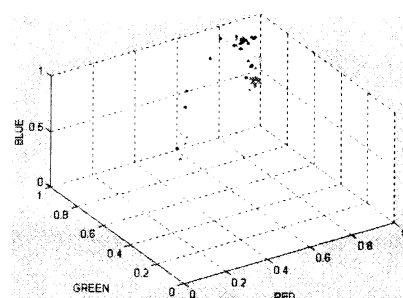


Figure 3. The 3-D mapping of Figure 2-(b)

We model this distribution by a look-up table. The look-up table is fast and can easily assign members that are scattered in space. The outputs of a look-up table show the tracker whether a member pixel is in the object or not. This approach can easily be extended to handle objects composed of various uniform color patches. Knowing which pixel belongs to which color, we can then do the look-up table with its constituent colors.

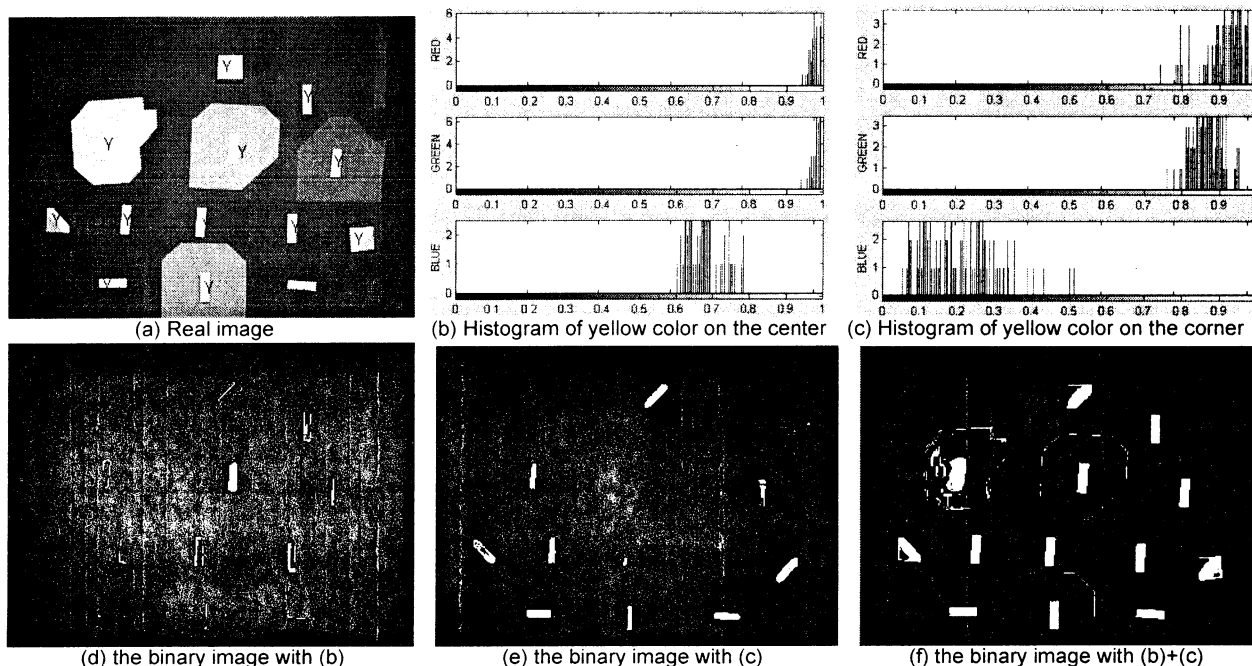


Figure 2

### 3. The Look-Up Table Approach and Experimental Results

In the proposed approach, there is one look-up table(LUT) associated with one color to be tracked. Because one pixel of the image is expressed by 16 bits, grabbing the length of one LUT is  $2^{16}$  bytes (= 64kbytes). The value of a pixel is used for indexing the LUT. One LUT formulates a model of the object's color in the given lighting conditions. This approach using LUT doesn't need any comparative statement or any operation. It is just one access to LUT. So this approach is faster than other approaches.

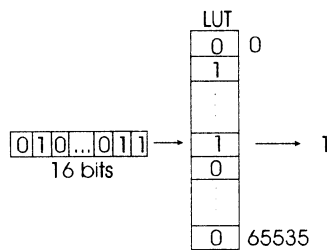


Figure 4. Processing of LUT

A tracker is a process that makes a binary image into a real image with LUT, labeling the blobs in the binary image, passing the size-filter, and computing the position and orientation of the blobs at every time instant.

This LUT method can pass the area with an odd shape. So the unnecessary area can be effectively removed in RGB space.

#### 3.1 Identification of LUT

The first step is supplying information to LUT information about the color of the object through a human operator. The operator indicates the object to be tracked by pointing at different parts of it in several spaces and fixes the boundary each time to record a color sample in the LUT. Each fixing of the boundary grabs the quadrangle that the operator assigns using a mouse. The values of LUT about the sampled pixels' color values are set to '1'. Other values of LUT are reset to '0'.

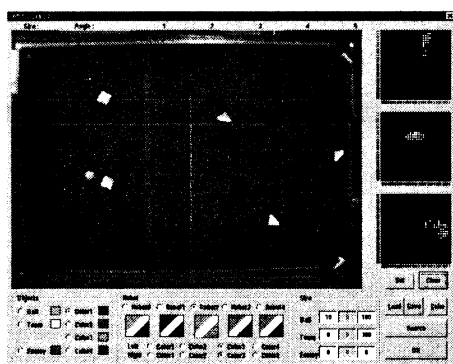


Figure 5. the appearance of developed program

This process is repeated until the LUT is completed over

the entire playing field. The operator just moves the objects and assigns the corresponding areas to that objects' color.

The R-G, R-B and G-B spaces in the screen of a developed program help this process.

#### 3.2 Adjustment of LUT

Though LUT is done over the entire game in the ground there are spots in several spaces of the binary image or in the object's shape which are not clear. These problems are caused by the boundary different colors. Figure 6 is the zoomed image on the boundary of a patch. The blending of the yellow with the black occurs in this part.

To avoid this problem LUT must not include blended colors. In the developed program the operator can zoom in on the boundary of the image by pointing with a mouse. Through the zoomed-in image the operator checks the value of the boundary pixels, and then adds or removes by pixel unit. This process adjusts the LUT and eliminates the troublesome spots.

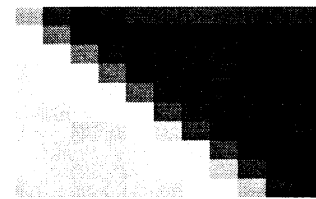


Figure 6. A zoomed in image on the boundary

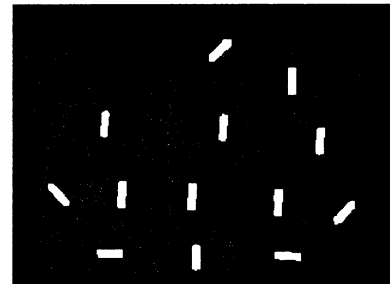


Figure 7. Processed Image with the proposed method about Figure 2-(a)

#### 3.3 Tracking the Robots and Balls

The tracking of the ball is very simple because there is no necessity for knowing the direction. The ball is the largest blob in the binary image, passing with LUT of an orange golf ball.

In the robots' case, the tracker is classified with the team color, which is yellow or blue, and then finds the ID colors. The identification of each robot is matched with the closest ID colors. The direction of the robot is computed by angle between team colors and ID colors. The resolution of the direction is about 3 degrees.

When the tracker loses an object, then the tracker assumes that the position of the object is the position of the previous frame. And then the tracker gives a warning sound. But in the case of making LUT over the entire playing ground, the tracker rarely loses the object. So in practice our tracking is highly reliable.

### 3.4 The results of the proposed experiment

In our system the access speed of a RAM about 320\*240 block is 5ms. The required time to track an orange golf ball is only 3ms, on the average. It required about 10ms to track the position and direction of three robots. The iteration to track three robots and a ball requires about 13ms. So we can catch the position of the ball and robots at 60Hz. Most of time is consumed in accessing the video RAM to get the image.

### 3.5 Further application for face image detection

Our method also works well for detecting human faces image under many indoors lighting conditions, both from a fixed as well as a mobile camera platform. Many previous works do not allow for camera movement. We have had good results with a hand held camera as shown in Fig. 8.

The binary image of Fig. 8 shows only the flesh in the image. We can easily detect the face to select the larger blobs.



Figure 8. Extracting flesh color in indoor lighting condition

## 4. Conclusion

In this paper we pointed out the disadvantages of the previous methods, and proposed a robust and fast approach in extracting the desired colors with LUT. We can easily track the position and orientation of three robots with a ball on a general PC at 60Hz with various illumination conditions. It is the maximum speed in a NTSC signal.

At present we are using the proposed method in a Narosot category in robotic soccer game. In this case the tracker finds the five robots and a ball. The proposed technique works well with the Narosot. We expect that the proposed method can be used not only for robotic soccer but also for other visual tracking application.

## References

1. J. Ivins and J. Porrill (1998), Constrained Active region models for Fast Tracking in Color Image Sequences, *Computer Vision and Image Understanding*, Vol. 72, No. 1, Oct., pp. 54-71
2. A. Smith and E. Lyons (1996), HWB - A more intuitive hue-based color model, *Journal of Graphics Tools*, Vol. 1, No. 1, pp. 3-17
3. R. Jain, R. Kasturi and B. Schunck (1995), *Machine Vision*, McGraw-hill
4. P. Golland and A.M. Bruckstein (1997), Motion from Color, *Computer Vision and Image Understanding*, Vol. 68, No. 3, December, pp. 346-362
5. C. Rasmussen, K. Toyama and G.D. Hager (1996), Tracking Objects By Color Alone, Technical Report at Yale University
6. K. Barnard, G. Finlayson, and B. Funt (1997), Color Constancy for Scenes with Varying Illumination *Computer Vision and Image Understanding*, Vol. 65, No. 2, February, pp.311-321

# The multi-agent system's design based on behavior-based learning model

Wang Xiaoshu\* & Masanori Sugisaka \*\*

Department of Electrical and Electronic Engineering, Oita University, 870-1192, Japan

\* E-mail: wangxs@cc.oita-u.ac.jp

\*\*E-mail: msugi@cc.oita-u.ac.jp

## Abstract

In this paper, as an ideal test bed for studies on multi-agent system, the multiple micro robot soccer playing system is introduced at first. The construction of such experimental system has involved lots of kinds of challenges such as sensors fusing, robot designing, vision processing, motion controlling and especially the cooperation planning of those robots. So in this paper we want to stress emphasis on how to evolve the system automatically based on the model of behavior-based learning in multi-agent domain. At first we present such model in common sense and then apply it to the realistic experimental system. At last we will give some results showing that the proposed approach is feasible to guide the design of common multiple agents system.

**Keywords:** multi-agent system, behavior-based learning, micro robot soccer playing system

## Introduction

With the deep going researches on kinds of control theories and increasing requirements of higher efficiency in industrial processing as well as in humming living, it is not surprising for not only the specialists but also common people in recent to pay so much interests on 'multi-agent system'. Here the agent is defined as anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors independently. Then, a 'multi-agent system' is one that is composed of more than two of such agents at least but is treated as an entirety during task resolving. Obviously such system can supply better performances, such as higher efficiency, stronger fault-tolerance capacity etc., than traditional monolithic system.

In spite of these advantages, how to drive multi-agent system to accomplish assigned tasks successfully within a dynamic environment is a very challenging problem and in fact up to now it is not ideal in building up such an applicable system.. The extreme difficulties mainly lie on the following two facts at least. At first, there are too much uncertainties not only among agents but also between the interior system with the extern environment. Especially in real time system, the behavior of any agent will change the surroundings continually. In usual these uncertainties are so complex and unpredictable that make it too difficult to coordinate agents both at the level of mechanics and of the control system well. On other hand, the errors and noise unavoidably exist in the sensor system, which is usually used to perceive the environment. Those error and noise keep us from mapping the sensors input to the optimal

output, especially in those system whose requirement to system real time ability is exacting.

Traditional controllers handle such problem by pre-wiring the special behaviors with corresponding sensors input in advance. But such an approach suffers from several problems. For example, the increasing complexity of the design process made it impossible for designer to consider all conditions the agent may encounter so that the pre-wired behaviors show unable to adapt to circumstances unforeseen at design time. Efforts to resolve these difficulties bring about the ideal of decomposition of control architectures and mechanism of self-learning in natural. Up to now a number of learning algorithms have been utilized for this purpose including kinds of neural networks, reinforcement learning, evolutionary algorithms and genetic programming and so on. Among them, the behavior-based systems proposed by Brooks have been praised for its robustness and simplicity of construction. Such systems are commonly composed of a collection of 'behavior-producing' modules that map environment states into low-level actions and a gating algorithm that decides, based on the state of the environment, which behaviors should be selected through and executed.

In this paper the behavior-based learning approach in multiple agent domains is proposed. What we want to do in end is to explore a feasible approach to guide the design of practical multiple agent system in common sense.

## Experimental System

As a realistic experimental system for the research on the multi-agent system, a kind of robot tournament system called *multiple micro robot soccer playing system* is constructed in our laboratory. Two robot teams, each of which must be composed of three micro robots at least, play the game. The rules are driven from the practical soccer sport played by humming beings. Up to now several relatively complete regulations have been drawn up by some international organization concerned such as FIFA. The overall system is shown in Fig.1. The reasons for us choosing such a system as test bed is for that it is (1) a standard multiple agent system suitable to be studied, (2) of flexibility. In other word, we can expand or contract the scale of system easily to meet the needs of ours. (3) easy to tell whether one kind of technology better than another or not, for it is a kind of antagonistic game played by two

teams.

A kind of wheeled mobile micro robot equipped with two wheels driven by DC motor independently is used. A 3 element vector  $\zeta \{x, y, \theta\}$  is defined to describe the robot's posture with respect to play field. Up to now no sensors are installed on the robot itself at all. The CCD camera serves as only sensors of whole system instead. What we want to note is that the speed of vision sampling and processing must be fast enough to meet the need of control in real time. In our real system, because of using a high performance VisionCaptureCard with the technology of DirectMemoryAccess, our system is capable of capturing vision frame from CCD camera with the sampling speed of 30frames/second. The software partition is developed using C++ language based on a personal computer with Intel Pentium 200 CPU.

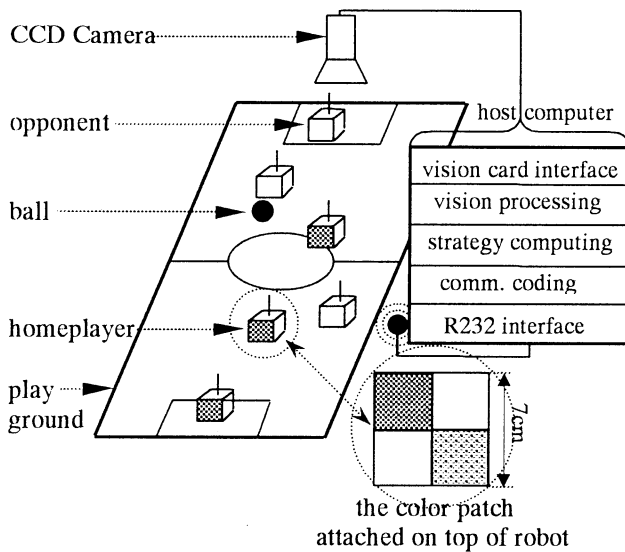


Fig.1 The system for micro robot soccer playing game

### Behavior-based learning model and its application on experiment system

The behavior-based model is shown I Fig.2 The partition encircled by the dot line show the interior system, which exits within a dynamic extern environment. The interior system perceives the extern environment through kinds of sensors it owning such as CCD Camera, Bump Sensors, Photosensors and so on and act on the extern environment by the actuators. During the system designing, the practical system scale, design objective and so on will decide which sensors should be chosen. As said above, in our experimental system up to now, only one CCD Camera is used for it can lighten the design overload and reduce the micro robot's dimension to conform the game rules. Because of kinds of sensors and noises exiting, the first thing we must do is SensorInput|Processing&Formation. According to the practical system, all kinds of techniques

can be used to extract correct signals from the noise-exiting background of sensors input.

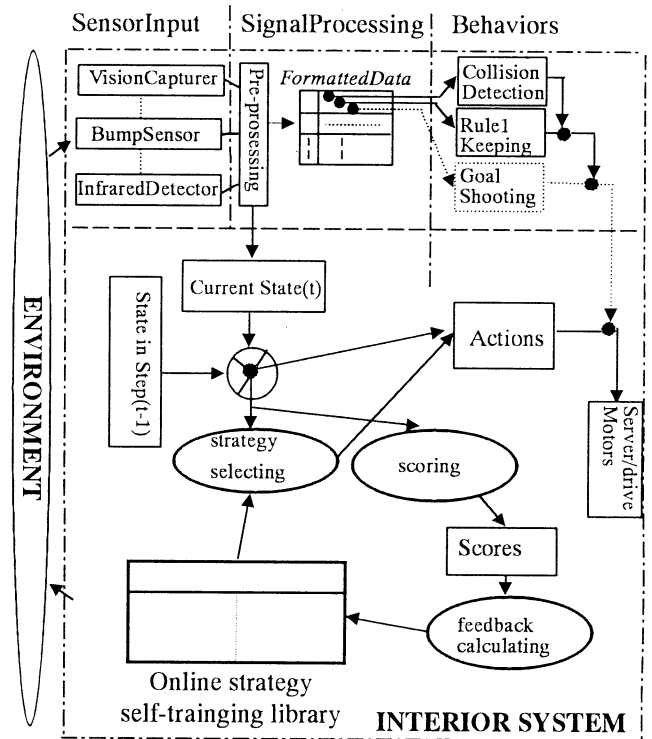


Fig.2. Behavior-based learning model

After we obtain formatted sensor data, the task becomes how to find a mapping from an n-dimension sensory space to a k-dimension motor space that optimizes the whole system's performance in terms of, for instance, doing more goals shooting. Because of the large dimensionality of sensory space as well as the motor space in such a multi-agent system, the task of hardwiring becomes extremely cumbersome and error-prone in realistic environment. In addition, trying to construct monolithic control architecture for the end aim of this system is obviously impossible. In our experimental system, these problems are addressed by decomposing the end task into layered elementary behaviors and performing these behaviors in the realistic actuators using the behavior-based approach.

Referring to the Fig.2, we divide the end task into two layers of subtasks. One is named *low-level reactive behaviors* shown on the top of the figure and another is known as *high-level goal-directed behaviors* shown on the below of the figure. The *low-level reactive behaviors* include three kinds of elementary behaviors. The first is called *spontaneous behaviors* such as *collision detection* and *obstacle avoidance*. The second is *ruleKeeping behaviors* so as to keep the robot from violating the game rules. For example, one of rule reads that "When more than one robot of the defending side enters the defense zone to substantially affects the game. The foul of Multiple Defense will be called, and the penalty kick will be declared." In



addition the *emergent behaviors* such as *goal shooting* and *goal keeping* are also be belonged to *low-level reactive behaviors* because these behaviors is crucial to winning the game. We can find that all of these *low-level reactive behaviors* share same, which are (a) they will be conducted by any robot not considering the relations with other robots, (b) they just are based on the subset of sensor data, (c) the reactions must be rapid. Based on above characteristic and requirement, in our current version we achieve these behaviors using simple pre-wiring strategy. In other word, we pre-decide how much the probability of a specified behaviors occurring is when a circumstance is encountered by robot. Such strategy is feasible for the subset the sensor space mapped to each behavior is relatively small.

Now we begin to explain how to achieve those *high-level goal-directed behaviors*. That is to say how to resolve the problems of cooperation and coordination among multiple robots. At first we encounter the problem of how to describe relations among all robots.

In order to meet the need of real time control and reduce the dimensionality of relation apace, we locate all robots respect with to the current position of ball. Please refer to the Fig.3. For instance, the robor1 is located at the section of B6. Then it is easy to find that in any time the maximal probability of location of each robot will not over 18. But considering the game played by two teams each of who includes three players, the probability of relations among these robots will increase up to  $18^6$ . It is obvious that it is still impossible to handle all of conditions at all. So an approach based on learning is proposed to address such problem. Please refer to the below partition of Fig.2.

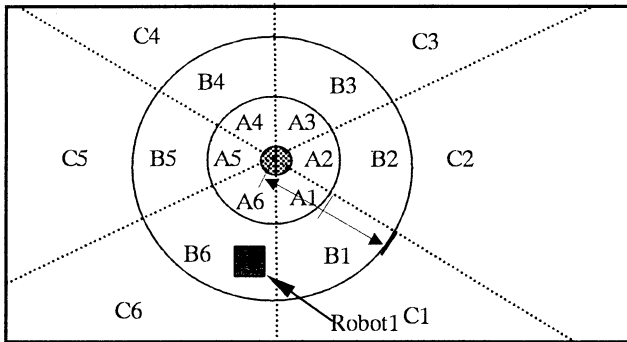


Fig.3. The location of objects in playground

At first we design a data structure named as *online training strategy library* shown in Fig.4. Each record composes 5 items, item1 records the number; item2 shows whether this record is valid, 1 means valid in current and 0 is for invalid record; item3 contains the current state of play field based on the location method shown above. \* means the specified object can not be identified in current vision frame or we do not concern the specified object at all; items4 shows where each robot in home team will go at next step. Referring to

the Fig.4.

0	1	A1C4B3C4**	B6	B4	*	55
1		-----	-----	-----	-----	-----
1000	0	-----	-----	-----	-----	-----
rec. No.	flag for used/not	current state	next pos.		score	

Fig.4. The online training strategy library

Considering the fact speed the robot can run, We decide that at one step a robot can only move to the section adjacent to its current position Then the robot1 in Fig.5 only can move to 4 probable sections in next step. Here the \* means the special object can not be found in current vision frame or we do not concern it at all; the last item is evaluation score for movements specified at item4 with the current state specified at item3. The range is from -100 to 100.

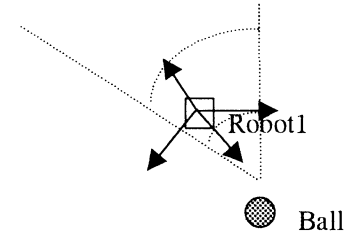


Fig.5. The movement of robot at one step

In order to achieve the learning mechanism three calculation modules called *strategy selecting*, *scoring* and *feedback calculating* separately. The procession of *strategy selecting* is a comparison and contrast one between the current state in play field with all valid records in the online training strategy library. The each comparison will result in a value denoted as resemblance value R

$$R = \sum_{i=0}^n p^i$$

Here  $P^i=2$  if the position of  $i_{th}$  home robot is located at the section specified by the  $i_{th}$  field in item3 or according field is \*. If not,  $p^i=0$ ; For the robot in opponent team,  $p^i=1$  if the match successes. The record with largest resemblance value will be selected to decide how the robot will go at next step. If the largest resemblance value is also equal 0, the position of all robots at next step will be decided randomly.

The procession of *scoring* is responsive to score the fact result following the movement followed by the *strategy*

*selecting*. In our experiment system, we just use a simple method to score such as: score  $S+=30$  if ball is kicked into opponent goal.  $S-=30$  if ball is kicked into home goal.  $S+=10$  if ball is controlled by robot of home team.  $S-=10$  if ball is controlled by robot of opponent team and so on.

In the end *feedback calculating* is used to adjust the training strategy library on line. The score of according record will be changed with the score calculated by the procession of score calculating. For a new condition, it will be added to library. Now the number of record in the library is kept not over 1000. In addition to those simple methods, some new functions for the feedback calculating such as merging, splitting is being added in order to converge more quickly.

### Experiment results

For doing experiment, at first we create an initial strategy library manually considering those typical conditions. Then we train such initial strategy library using a simulation system. The three simulated robots driven by the control method shown above play the game with other three robots controlled by man. After we obtain a relatively ideal result. We transfer the online training strategy library to a real experiment system. Now the three real robots driven by the control method shown above play the game with opponents driven by our first version control method. To our surprise, after some time of playing the robots driven by new control method do some cooperation behavior such as that shown in Fig.6. Fig.7 shows the ration of goals obtained by new version system to that by first version system.

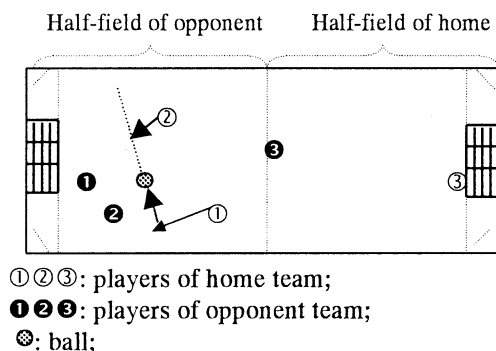


Fig 6. The typical attack cooperation

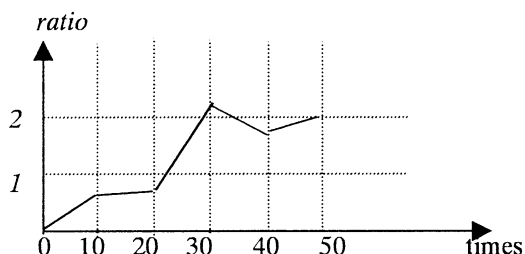


Fig.7. The ration of goals obtained by new version system to that by first version system.

### Conclusions

In this paper, we proposed the behavior-based learning model to guide the design of multi-agent system in common sense. By dividing a task into several layers of control and partitioning the sensorimotor space in a hierarchical behavior-based manner, the problem of high dimensionality of the sensorimotor space is alleviated. For the achievement of *high-level goal-directed behaviors*, the learning approach based on *online training strategy library* is proposed. Such model is applied on a realistic experimental system of *multiple micro robot soccer playing system*. The results show that such model is practically feasible better than the traditional pre-wiring methods.

The proposed method also has certain drawbacks that remain a subject of future research. For instance, *the low-level reactive behaviors* are achieved still using pre-programming methods. Learning model such as neural network or Q-learning will be tried to emerge these behaviors autonomously. In addition, the procession of *scoring* and *feedback calculating* is too simple yet and the online training strategy library is of unsteady. All these will be resolved in our next version system.

### References

- [1] M.Sugisaka and X.S.Wang and J.J.Lee: "The genetic algorithms(GAs) to evolve multiple agent cooperative system" Proceedings of the 3rd international symposium on artificial life and robotics. Beppu, Japan, 19-21 Jan. 1998. pp178-182.
- [2] M.Sugisaka and X.S.Wang: "The autonomous evolution of cooperation activity of micro-robots by genetic algorithms(GAs)" Proceedings of the 29th ISCIIE international symposium on Stochastic system and its applications. Tokyo, Jaoan, 10-12 Nov. 1997. Pp.29-32.
- [3] Tien C. Hsia and Michael Soderstrand: "Development of a micro robot system for playing soccer games", Proceedings of KAIST MIROSOT'96, Taejon, Korea, 9-12 Nov. 1996, pp.149-152.
- [4] Sahota, Mackworth Kingdon and Barman: "Real-time control of soccer-playing robots using off-board vision", IEEE Intern. Conf. System, Man and Cybernetics, Canada, 22-25 Oct. 1995, pp. 3090-3.
- [5] K.Ohkawa, T.Shibata and K.Tanie: " Self-generating algorithms of evolution for cooperative behavior among distributed autonomous robots", Proceedings of AROB'97, Beppu Oita, Japan, 18-20 Feb. 1997, pp123-126.
- [6] Guy Campion, Georges Bastin and Brigitte D'Andred-Navel: "Structure properties and classification of kinematic and dynamic models of wheeled mobile robot", IEEE transactions on robotics and automation, Vol.12, No.1. Feb. 1996, pp47-61.

# An Intelligent Control Strategy for Robot Soccer

Tae-Yong Kuc, Seung-Min Baek, In-Jae Lee, and Kyung-Oh Sohn

Intelligent Control and Dynamic Simulation Lab.  
School of Electrical and Computer Eng., Sung Kyun Kwan University  
Chunchun-dong 300, Changan-gu, Suwon, Kyungki-do 440-746, Korea  
E-Mail : tykuc@yurim.skku.ac.kr

## Abstract

This paper presents an intelligent control strategy for coordination of soccer playing mobile robots. Within the framework of hierarchical control structure, three layered components of supervisor, coordinator, and executer emulate the basic three concepts of human intelligence, perception, reasoning, and learning. Moreover, in the actual environment of robotic soccer game, these three concepts are implemented in the form of visual perception, state-action reasoning, and dynamic input learning for motion control of multiple mobile robots for successful soccer game. Computer simulation and experiment for real soccer game demonstrate that the proposed intelligent control strategy is feasible and effective in terms of structural simplicity and computational speed for real-time control.

**Keywords-** robot soccer, hierarchical control, perception, reasoning, learning

## 1 Introduction

The robotic soccer has recently been receiving considerable amount of attention not only as an interesting entertainment but also as a dynamic test bed for which many coordination techniques for coordination and motion control multiple mobile robots can be applied and evaluated in terms of feasibility and effectiveness. The robotic soccer systems and control schemes have been also developed by many researchers with various degrees of application and success<sup>[2,11,13,14,15]</sup>. However, most of the previous control strategies developed so far for robotic soccer system have been tested with only computer simulation due to high demand in real-time computing power or heavy communication link which is beyond the reach of current technology. In this paper, we propose and implement an intelligent control strategy for robotic soccer system which is structurally simple and computationally fast and hence can be implemented in real-time.

In general, an Intelligent control architecture for

robot soccer system is mainly concerned with skilled coordination of multiple mobile robots for successful soccer game like as a mastery of trained human soccer players with dexterity. On the other hand, the components and integration of intelligent robotic soccer system have as many variances as there are researchers in this area. For example, one can construct a centralized or decentralized controller for robotic soccer system, behavioral, representational, or hybrid architecture for soccer system performance<sup>[1,3,4,5,6,7,12,16]</sup>. However, from the conceptual view of intelligent control system, an intelligent soccer robot has to implement all or some of the three ingredients of human intelligence : perception, reasoning, and learning. The intelligent control strategy proposed in this paper tries to emulate these basic concepts of intelligence which appears in the form of visual perception of working environment, state-action reasoning, and learning of state transition model and dynamic input. The hierarchical controller with the intelligent control strategy is also implemented in the experimental setup of centralized robotic soccer system developed for MIROSOT<sup>[8,9]</sup>. It is well-known that the general drawback of centralized coordination system for environment of multiple mobile robots is heavy computational load for real-time control. To overcome the drawback of centralized control method, we introduce a simple hierarchical architecture as an intelligent robotic soccer controller with three layers of hierarchies which are represented by supervisor, coordinator, and executer, respectively. In the intelligent controller architecture, the supervisor provides global supervision with the ability of perceiving environment and discriminating the objects from visual sensor data. A simple high-level learning strategy in reasoning from the perceived information is implemented in the coordinator whose simplicity makes it possible to overcome the general drawback of heavy computational burden in using the centralized coordination approach for motion control of multiple mobile robots. The basic idea is in accelerating the speed of computation by selecting only a small number of discrete states from the configuration space based on which reasoning of state transition model and desired action selection are performed. The executer controls

the mobile robots for successful soccer game by providing desired trajectories which achieves the learned state transition model represented as the desired state-action pairs in the central coordinator. As an individual strategy, action level learning scheme for movement of each robot is also activated in the executer which learns the inverse dynamics input of each mobile robot to track the given desired trajectory for successful soccer game. Therefore, there exist two learning strategies in the intelligent controller architecture: the perception level or high-level learning strategy and action level or low-level learning strategy. Figure 1 depicts the overall configuration of the proposed intelligent controller for robotic soccer system. In the figure, three basic concepts of human intelligence, perception, reasoning, and learning, are implemented in the form of visual perception, state-action reasoning, and servo-level input learning at the three layers of control hierarchy. It is also found that the presented intelligent control architecture has properties of from the top to bottom transition of high- to low-level of intelligence and low- to high-resolution of sensor data and representation of model. The feasibility and effectiveness of the proposed intelligent control strategy with the simple hierarchical architecture is demonstrated via computer simulation and experiment for real soccer game.

## 2 The Intelligent Control Strategy

As a supervising controller, the supervisor provides the global information of the working environment from the visual sensor data, while the centralized coordinator for robotic soccer system plays a role in making the soccer robots work and collaborate for successful soccer game. In the coordinator, the state-action reasoning, learning of state transition model, and generation of motion trajectories take place for soccer robots to achieve the desired state in the planar configuration space.

### 2.1 High-Level Reasoning and Learning Strategy

In this subsection, we provide a simple reasoning and learning strategy for the coordinator which overcomes the general drawback of heavy computational burden in using the centralized coordination approach for motion control of multiple mobile robots. The basic idea in accelerating the speed of computation is to choose only a small number of discrete states from the configuration space and let the most efficient state transition model be activated first. After a desirable degree of freedom (robot) being learned which is the fittest one for reaching the possible goal position, an action command from the action set is defined in the direction that the action selected results in the desirable next state for successful soccer game.

**2.1.1 Configuration of Soccer Robots:** Assume that all the objects in the planar workspace are identified which include the ball, robots, and the goal post. In defining the states for each mobile robot or each degree of freedom in the configuration space, we consider the following two configurations based on the relative position of robot to the positions of opponent goal post and ball: *positive(desirable)* and *negative(undesirable)* configurations. In Fig.2, the positive or desirable configuration is defined as the configuration of robot such that the projection of robot to the extended straight line from the opponent goal post to the ball lines up in the order of the goal post-ball-robot(GBR). On the other hand, Fig.3 shows the negative or undesirable configuration in which the projection of robot to the linear line in between the opponent goal post and the ball makes line-up in the order of the goal-robot-ball(GRB). Note that the positive configuration means that the robot is demanded to move forward directly to the ball to make the ball closer to the opponent goal position, while the negative configuration is not. Based on the two configurations considered, we define a symbolic representation of positive states, negative states, and the singular states for configuration of each mobile robot.

$$S_P = \{GBRc, GBRr, GBRl\}$$

$$S_N = \{GRBc, GRBr, GRBl\}$$

$$S_S = \{GB = Rr, GB = Rl\}$$

where  $S_P$ ,  $S_N$ ,  $S_S$  represent the set of positive states, negative states, and the singular states, respectively. As shown in Fig.4 - Fig.7, the discriminating suffixes  $c$ ,  $r$ , and  $l$  stands for the robot on the center, right plane, and left plane of the straight line between the goal post and the ball in the direction of the ball facing the goal post. In the singular set,  $GB = R$  means the projection of robot to the  $G - B$  line meets the ball. Note that the singular states are the states which do not belong to the sets  $S_P$  or  $S_N$ . To provide obstacle avoidance action for mobile robot when necessary, the obstacle avoidance state can be defined by simply inserting an obstacle( $O$ ) in between the ball( $B$ ) and robot( $R$ ) as in Fig.8. That is, the sets of obstacle avoidance states  $S_{PO}$ ,  $S_{NO}$ ,  $S_{SO}$  are given as

$$S_{PO} = \{GBORc, GBORr, GBORl\}$$

$$S_{NO} = \{GROBc, GROBr, GROBl\}$$

$$S_{SO} = \{GB = O = Rr, GB = O = Rl\}$$

Therefore, all the states sum into 16 which is quite small number of states for the centralized coordinator. Note also that if  $n$ -dof or  $n$ -robots need to be coordinated simultaneously, the centralized coordinator have to treat  $16n$  states in the state search space, where in our case of robotic soccer system  $n$  equals 3.

### 2.1.2 Learning of State Transition Model:

In order to transit the current state to the most desir-

able next state for successful soccer game, the coordinator should learn to which state the current state is transferred, i. e., which robots to move. The simple criteria used in learning of the state transition are as follows:

- i) transit the negative, singular, and obstacle avoidance states to one of positive states.
- ii) in selecting one positive state, choose the most similar state to the current state in terms of moving distance and angle by one degree of freedom(robot).
- iii) choose the most desirable state in terms of the distance and angle of approach to the ball.
- iv) choose the most desirable state with robot with the minimal deviation from the  $G - B$  line.

The state transition model can be learned off-line using computer simulator and/or experimental setup and applied to the real soccer game. If necessary, the learned state transition model can be also modified through on-line learning of real soccer game.

### 2.1.3 State-action Reasoning:

Since the state transition model learned dictates a desirable direction of movement for each robot, the action can be chosen for each state from the action set defined below. In deriving an action set for movement of mobile robot, the robot motion is confined with only the following actions for simplicity : attack( $A$ ), wait( $W$ ), block( $B$ ), and compete( $C$ ). Hence, the symbolic representation of actions in the action set( $S_A$ ) is given by

$$S_A = \{A, W, B, C\}$$

The following three examples describe the procedures of state-action reasoning and action determination during a soccer game.

- i) If the states of robot 1( $R1$ ) and robot 2( $R2$ ) are positive and  $R1$  is closer to the ball than  $R2$ , then  $R1 = A$  and  $R2 = W$ .
- ii) If  $R1$  is in singular or negative state and  $R2$  is in positive state, then  $R1 = W$  and  $R2 = A$ .
- iii) If the states of  $R1$  and  $R2$  are negative and the ball is located at the defence zone and closer to the goal post than a predetermined distance, then  $R1 = B$  and  $R2 = B$ .

### 2.1.4 Path and Velocity Trajectory Generation:

If an action command is chosen from state-action reasoning using the learned state transition model, target position and velocity trajectories of each mobile robot are determined for the next desired state which are dependent on the maximum possible running speed of each robot. The target positions are also dependent on the action command chosen. For example, the target positions of a mobile robot for the attack( $A$ ) and wait( $W$ ) commands at the instance  $k$  are determined as follows:

$$PA_{k+1} = (xa(k+1), ya(k+1)) = (B_{x(k)} + \delta_a, B_{y(k)})$$

$$\begin{aligned} PW_{k+1} &= (xw(k+1), yw(k+1)) \\ &= (B_{x(k)} + \delta_w, \delta_l \text{ if } B_{y(k)} \geq y_0 \text{ else } \delta_r), \end{aligned}$$

where  $x$ -axis is the direction of attack,  $(B_{x(k)}, B_{y(k)})$  is the ball position,  $y_0$  is the  $y$  position of center of ground, and  $\delta$ 's are predefined real numbers via experiment which depend on the mobility of mobile robots.

Similarly, the maximum value of target velocity is determined according to the action command chosen. The highest ones for attacking motions of shooting and dribbling and wait transition, the medium one for the compete command, and the slowest one for the blocking motion. In order to servo the mobile robot to the target position with the chosen maximum velocity, the most efficient path and velocity trajectories are generated in the coordinator for the servo control of mobile robot by the executer. In general, the shortest path is designed under the dynamic environment of robotic soccer playground. However, the generated path and velocity trajectory are not necessarily optimal ones in terms of safety and time due to moving obstacles of opponent robots.

## 2.2 Servo Level Learning of Mobile Robots

Since precise servoing of the target path and velocity trajectory is not guaranteed with only the conventional PID controller, a servo level dynamic learning control strategy is activated in the executer as a viable alternative to the PID velocity servo control method. It is well known that a dynamic learning controller enhances the robustness of robot motion control to various uncertainties such as in the weight of the ball, the friction coefficient of ground, parameters of kinematics and dynamics of robot, etc.. Now, let the dynamics of mobile robot be represented by

$$W(\eta, \dot{\eta}, \theta) = \tau,$$

where  $\eta \in R^2$  is the angular and linear velocity of mobile robot,  $\tau \in R^2$  is two dimensional torque input vector and  $\theta \in R^l$  is the unknown parameter vector of mobile robot. The desired torque input is learned off-line via a set of experiments or on-line during the real soccer game and converted into the voltage input for each mobile robot actuated with DC motors<sup>[10]</sup>. Moreover, the learned information can be implemented using the soft computing tools such as CMAC or a radial function basis NN. Then, the uncovered region in the training phase is interpolated by the universal approximator.

## 3 Conclusion

In formulating the intelligent control strategy for robotic soccer players, we implicitly have assumed the opponent plays the soccer game reasonably well and

can move as fast as our robot. Through computer simulation and experiment, the computationally simple control strategy is validated by training the robotic soccer system using the proposed intelligent controller which emulates the capability of human soccer players, environmental perception, state-action reasoning for team strategy, and skilled motion learning for individual strategy.

## References

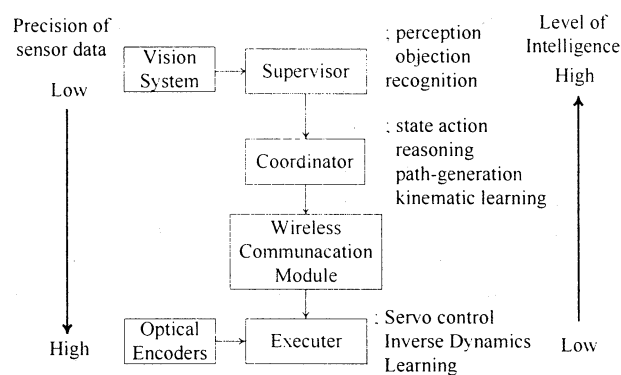
1. Arkin R.(1992), Cooperation without Communication:Multi-agent Schema Based on Robot Navigation, Journal of Robotic System, Vol. 9, No. 3, pp. 351-364.
2. Asada M., Noda S., Tawaratsumida S. and Hosoda K.(1995), Vision-Based Reinforcement Learning for Purposive Behavior Acquisition, Proceeding of the IEEE Conf on R.&A. , pp.146-153.
3. Barraquand J. and Latombe J.C.(1991), Robot Motion Planning : A Distributed Representation Approach, Int. J. Robot. Res., Vol.10, pp628-649.
4. Beni G. and Wang J.(1991). Theoretical Problems for the Realization of Distributed Robotic Systems, Proc. of ICRA-91, Sacramento, CA, pp. 1914-1920.
5. Brooks R.(1986), A Robust Layered Control System for a Mobile Robot, IEEE Trans. on Robotics and Automation, RA-2.
6. Byrd J.S.(1993), Computers for Mobile Robots, In: Zehng Y.F. (ed) Recent Trends in Mobile Robots, World Scientific Pb LTD, pp. 185-210.
7. Latombe J.C.(1991), Robot Motion Planning, Kluwer Academic Publishers, Boston, USA.
8. Micro-Robot World Cup Soccer Tournament (MIROSOT'96), proceedings, KAIST, Daejon, Korea.
9. Micro-Robot World Cup Soccer Tournament (MIROSOT'97). Call for participation, KAIST, Daejon, Korea.
10. Nam J., Baek S. and Kuc T.(1996), A Robust Nonlinear Optimal Controller for Autonomous Mobile Robots, IEEE Int. Conf. on SMC, Vol.2, pp.1453-1458.
11. Noda I.(1995), Soccer server : a simulator of robocup, Proceeding of AI symposium '95, pp.29-34.
12. Reif J.H. and Wang H.(1994), Social Potential Field : A Distributed Behavioral Control For Autonomous Robot, Proceedings of the 1st Workshop on the Algorithmic Foundations of Robotics, MA, USA.

13. Sahota M.K.(1993), Real-time intelligent behavior in dynamic environments: Soccer-playing robots, Master's thesis, University of British Columbia.

14. Sohota M.K., Mackworth A.K., Kingdon S.J., and Barman R.A.(1995), Real-time Control of Soccer Playing Robots Using off-board Vision : the Dynamic Test-Bed, Proceeding of the IEEE Conf on R.&A., pp.3690-3693.

15. Stone P., Veloso M., and AchimS.(1996), Collaboration and Learning in Robotic Soccer, Proceeding of the Micro-Robot World Cup Soccer Tournament, pp.26-37.

16. Wang J. and Prevuti S.(1995), Distributed Traffic Regulation and Control for Multiple Autonomous Mobile Robots Operating in Discrete Space, Proceeding of the IEEE Conf on R.&A., pp.1619-1624.



**Figure 1:** Controller Schematic of Robotic Soccer System

# The Design and Development of Robot Soccer Player

Chunhui Zhang    ChenQu    DaZhi Gao

Control & Simulation Research Center (P.O.Box 128)  
Northeastern University, Shenyang, 110006, P.R.China  
Email:ctrlsim@mail.neu.edu.cn

**Abstract:** Basically, robots, a vision system, a host computer and a communication system are needed for soccer robot system. In this paper, we present the hardware and software's design and development of NewNEU team's player. A soccer game of MIROSOT played by 3 robots from each team is a lively research mode of multi-robots and artificial life, and the performance of robots determines the performance of whole system to much extent.

**Key Words:** robot soccer, wheel type robot, obstacle avoidance, FSK, PWM,ICR

## 1 Introduction

Robot soccer, referring to the fields of robotics, intelligent control, data fusion, computer technology, wireless communication, image processing, artificial life, mechanics and so on, is a good test bed for multi-agent system and multi-robot theory<sup>[1]</sup>. Internationally, more and more researchers pay attention to it.

Basically, the whole system consists of four subsystems: robots, vision, decision-making and communication subsystem. According to MIROSOT (Micro-Robot World Cup Soccer Tournament), 3-5 robots from each team whose complete size are within 7.5cm<sup>3</sup> play soccer game as figure 1<sup>[2]</sup>.

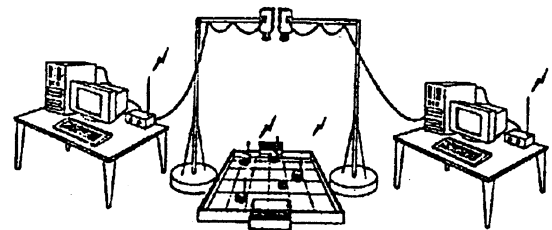


Figure 1

The control scheme of NewNEU team is a vision-based one, which controls or manipulates the robots by processing information, which includes robots and ball's posture from vision system with a host computer and sends command to robots directly. The command can control robots to accomplish a serial of action such as shoot and block.

The vision system of NewNEU use color CCD camera and Peking KJK Company's image-grab card, which can process 25 frames image per second. The decision-making system utilizes six-step inference mode by Pro.Xu Xinhe. The robots receive the left and right speed setpoint commands from decision-making system, accomplish speed control and simple obstacle avoidance.

This paper presents the design and development of soccer robots of NewNEU, and try to implement it quickly in a low cost, and make it popular.

## 2 The hardware design of player

The player mainly consists of CPU board, communication module, driving mechanism, sensor, etc. The body of player is composed of 3 PCBs with surface mounting devices, and which are firmed by four metal bar. The upper PCB is CPU board and communication module, the middle PCB is driver board, and the bottom PCB is used to settle power and motors. For MIROSOT, the robots should satisfy the following demands: flexibility, smoothing, robustness, maintenance, low cost and the complete size within 7.5cm<sup>3</sup>.

### 2.1 Movement part

For the moving mechanism, a wheel type or a cater pillar type<sup>[1]</sup> can be considered. The cater pillar type robot has features as follows: exact straight motion, robust to slipping, inexact modeling of turning. Omnidirectional robot<sup>[3]</sup> using three wheels has features as follows: free motion, complex structure and weakness of the frame. Among the wheel type, the bi-wheel type robot is suitable to make the model, smooth motion and risk of slipping.

NewNEU's player uses bi-wheel type structure which belongs to ICR (Instantaneous center of rotation), and two wheels' axes cross one point as figure 2(a). As

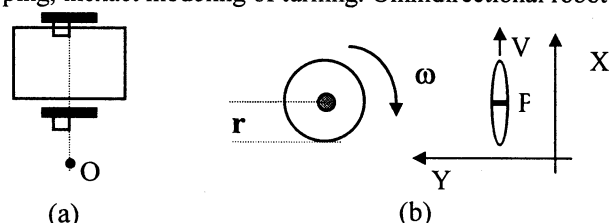


Figure 2

figure 2(b) the wheel is fixed, so the point P cannot move to the direction perpendicular to plane of the wheel. On the assumption that non-slipping and pure rolling would occur between the wheel and ground, the velocity of point P:

$$V = (r \times \omega) E_x \quad (1)$$

where,  $E_x$ : a unit vector to x axis.

The degree of mobility of this structure is 2, and the degree of steer ability is 0 (there is no centered orientable wheel that can be independently to steer the robot).

The posture of the robot:  $P = [x_1, y_1, \theta]^T$ , where,  $(x_1, y_1)$ : position of the robot;  $\theta$ : orientation of the robot. The control input:  $U = [V, \omega]^T$ , where,  $V$ : linear velocity of the robot;  $\omega$ : angular velocity of the robot. Furthermore, we get the relation between the control input and speed of wheels:

$$V_L = r \omega_L \quad V_R = r \omega_R \quad (2)$$

$$W = (V_R - V_L)/L \quad V = (V_R + V_L)/2 \quad (3)$$

where,  $V_L, V_R$ : linear speed of left and right wheel,

$\omega_L, \omega_R$ : angular velocity of left and right wheel;

$L$ : the distance between two wheels.

The kinematics equation of bi-wheel type robot is presented as follows:

$$\begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{\theta}_c \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (4)$$

This structure belongs to ICR as figure 3, so the angular velocities of left and right wheel are equal. The following shows the basic motion control,

$$\frac{V_R - V_L}{L} = \frac{V_R}{R + L/2} \quad R = \frac{L(V_R + V_L)}{2(V_R - V_L)} \quad (5)$$

where,  $R$ : radius of rotation; when  $R = \text{Infinity}$   $V_R = V_L$ , straight motion;  
 $R = 0$   $V_R = -V_L$ , rotation motion.

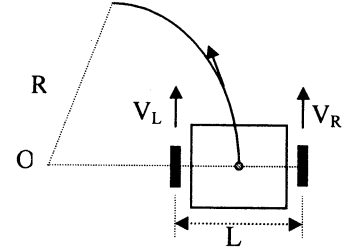


Figure 3

## 2.2 Driving part

In order to be able to move over the whole playing field quickly, each robot has to be able to move straight and to turn rapidly. Therefore, each player carries two motors and each motor drive a wheel on one side of the robot.

As actuators, step motors and DC motors can be considered. The speed loop would not be needed<sup>[4]</sup> if using step motors, but mini DC motors are more available. Torque, speed and power consumption is important in choosing motors. To make small robots, it is necessary to get the one chip motor controller along with the motors, which have a gear train and an encoder together, however, these kind of motors and their controller are not available or too expensive in domestic. So NewNEU team uses the motors and gear train from toy car, and use self-made circuit for motors driver.

Speed and direction control of motors is needed. To change motor's volts polarity, we should use H-bridge circuit, use PWM (pulse width modulation) methods to change the average volts of motors, so as to control the speed of motor.

Figure 4 shows the driver circuit and PWM control method, where, DIR: direction control signal input, PWM: input signal of PWM pulse. When DIR is '1', the motor rotates positively. In reverse, the motor rotates negatively when DIR is '0'. The triodes of BD433, BD434 are used, so

this driver circuit can provide the motor with voltage of 0-3V and max current of 2A.

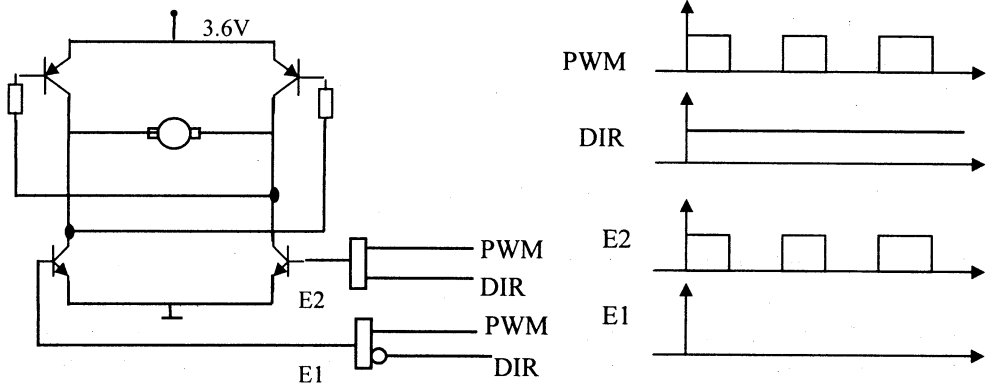


Figure 4

## 2.3 Communication system



According to MIROSOT rules, the data, which comes from host computer, must be sent to the robot by an wireless communication. So IR-remote controls or RF-digital communication system is necessary. In an IR-remote control method the circuitry is relatively simple when the number of channels is less, but as the number increases it will become considerably complex<sup>[1]</sup>. So we use the RF-digital communication system.

### 2.3.1 Modulation mode

The modulation mode used for RF-digital communication includes ASK, FSK and PSK. In ASK mode the system circuitry is very simple, but it is easy to be interfered by environment noises. In FSK mode, the system may have very well anti-jamming capability. Moreover, there are many kinds of integrated chips for choosing, such as MC2831 MC2833 transmitter chip and MC3356 MC3362 receiver chip.

### 2.3.2 System scheme

The transmitter centers on MC2833. The frequency of 36.00MHz and 42.95MHz can be chosen and its modulating sensibility is 10Hz/mv when modulated voltage is 1v. Since the crystal oscillation is used to make frequency stable, the stability degree of frequency can reach to  $10^{-5}$ . The transmitter's function is illustrated by figure 5.

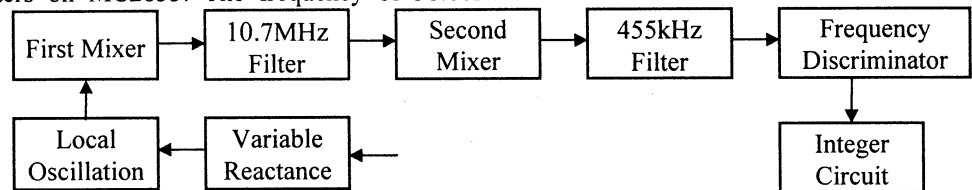


Figure 5

MC3362, which is a powerful chip of modulating frequency receiver, is the center of the receiver. As figure 6, mixers, frequency discriminator and shaped circuit are showed. The high frequency signal is transmitted into the chip through the matching network, then the signal be mixed with the first local oscillation so as to produce 10.7MHz intermediate frequency which is send into filter. Then the intermediate frequency of 455KHz is generated through two-level mixer; Finally, the signal formed is send into frequency discriminator in order to demodulate out digital pulse signal, which must be trimmed in order to be able to be transmitted into singlechip

Since the narrowband receiver requires very high stability to the local oscillation frequency, in the experiments the first level local oscillation always have some small change of frequency in the fifteen minutes so that receiver can not work normally. Adding a phase-locked loop circuit can solve this problem. With this circuit, the frequency can be tuned automatically.

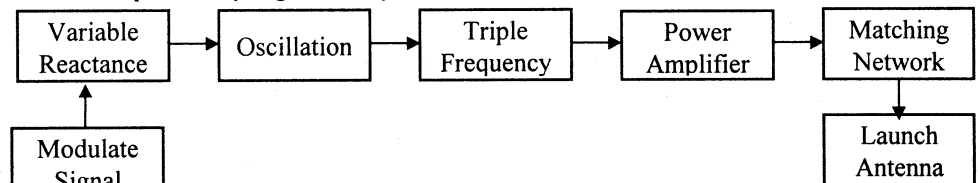


Figure 6

## 2.4 The design of CPU board

To realize high intelligence, a high performance CPU is necessary. Considering the limitation on size and power consumption, a one-chip controller with versatile functions will be suitable<sup>[1]</sup>. Choosing a suitable scheme, you can get high performance-price ratio.

Scheme first: high performance CPUs, such as Intel 16bit processor 80c196kc(20MH)<sup>[2]</sup> with 3 PWM ports, 8 A/D channels and 2 counter, which have more capability of computation and extension, are good candidates. They can accomplish speed and position control, obstacle avoidance and other complex functions with peripheral device.

Scheme second: common CPUs, such as 8051 compatible products, with some special use chip such as LM629 (precise motion controller), can accomplish PID control to motor.

Scheme third: choosing some special singlechip, such as 8XC51FA compatible products, with 5 PCA(programmable count array), which can generate PWM pulse and count feedback pulse from encoder.

According to field and interest, the designer can choose his own scheme. Scheme first has capability of upgrading and increasing its intelligence, and is suitable to be research mode, but it may cost much. Scheme 3 has high-integrated level, and saves the space of player. However, scheme 2 is more suitable to play game with higher performance-price ratio.

## 3 Software design

### 3.1 Communication protocol

Currently, the robot control words consist of 3 parts: robot ID, command part and data part. The command part

specifies the mode of robot actions, and the data parts specifies how fast and how far the robot should move<sup>[5]</sup>. According to the difference of intelligence level of player, the complexity of command is not same. If the robot has higher intelligence, the host computer can send action command. If the robot has lower intelligence, it can only control speed, just like remote control car<sup>[6]</sup>, then the command would be include left and right wheel speed.

NewNEU team is centralized control system. The player, which can accomplish speed control and simple obstacle avoidance, vision system and host computer can compose a feedback system. The protocol is presented as right table, where, upper nibble of 1111 is a header byte to validate the data receives.

Byte1				Byte2	Byte3
1111	M	X	ID	Left Speed	Right Speed

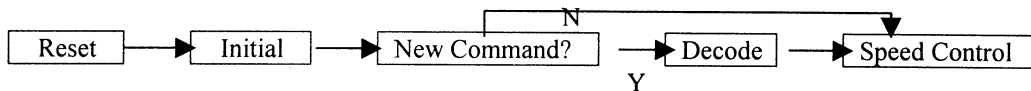
M: mask IR sensor flag,

X: reserved bit,

ID: 2-bit player's identification.

### 3.2 Program design of on-board CPU

The flow chart of the main program of robot is showed as follows.



Command received program and obstacle avoidance program are work in interrupt way. Simple actions for avoiding obstacle are preferable for rapid reaction, and obstacle avoidance algorithm is showed as follows.

If { LeftFront Obstacle } Then QuickTurnRight(θ);

If { RightFront Obstacle } Then QuickTurnLeft(θ);

If { Both side Obstacle } Then QuickTurnRight(θ)。

### 4 Conclusion

Soccer robot is a multi-subject intersection field. Robot soccer game is a very complex robot application, which provide us with a lively research mode for multi-agent, multi-robot and artificial life theory. Moreover, the player incorporates robot control, embedding system, data fusion and wireless communication technology, etc. That would attract more researchers' attentions.

### References

- [1] J.-H.Kim,"Cooperation Multi-Agent Robot System:From the Robot-Soccer Perspective", Proc. MIROSOT'97,pp.5~16,June,1997.
- [2] H.-S.Shim,M.-J.Jung,H.-S.Kim,I.-H.Choi and J.-H.Kim,"Development of Vision-Based Soccer Robot System for Multi-Agent Cooperative System ", Proc.MIROSOT'97, pp.29~36,June,1997.
- [3] R.Blank,J.-M.Koller,M.Lauria,F.Conti,"Omnidirectional Robot and Fast Positioning Systems",Proc.MIROSOT'97,pp.85~86,Nov.1997.
- [4] Y.-F.Seng,"NP Micro-Soccer Robot:Design and Implementation", Proc. MIROSOT'96, pp.95~103,Nov.1996.
- [5] K.-H.Kim,"Multiple Micro Robots Playing Robot Soccer Game",Proc. MIROSOT'96, pp.38~43,Nov. 1996.
- [6] D.-Y.Kim,"Development of Multiple Mobile Robots Playing Soccer",Proc. MIROSOT'96 , pp.83~86,Nov.1996.

## Simulation Model of Micro-robot Soccer System

FengRui Xiaodong Liang Xinhe Xu

Control and Simulation Research Center (P.O.Box 128)  
Northeastern University, Shenyang, 110006, P.R.China  
Email:ctrlsim@mail.neu.edu.cn

**Abstract:** In this paper, we set up all the objects' models in the robot soccer system by applying the technique of system modeling and computer simulation. Based on the models, we erect the MIROSOT simulation system, so as to provide a effective tool for studying the strategy system of robot soccer and a confront flat roof on software of the computer simulation robot soccer.

**Key Word:** robot soccer, system simulation, system modeling

### 1. Introduction

Robot soccer introduced here is a sort of miro-robot soccer, the abbreviation is MIROSOT, both teams send three or five robots, whose size limits on 7.5cm\*7.5cm\*7.5cm, in the game. So it is a cooperative Multi-Agent system. The system includes four subsystems: vision system, strategy system, robot system and communication system. Robot soccer based on the computer simulation is similar to MIROSOT, the only difference is that both teams confront on the computer screen by software, so it become available that many teams, which haven't robots, also can join the game.

System simulation has many good qualities such as rapid, economy, easy to carry out, and so on, so it should be a main tool in the research of the system. According to system simulation, we can checkout the validity of the strategy arithmetic, develop new strategy, and provide a confront flat roof on software. As the real system is very complex, the moving states of the robots and ball are unknowable, so we must set up models to describe them. The main function of the MIROSOT simulation system is to describe all objects' moving state in the ground by setting up all the objects' dynamics model.

The simulation system we designed has the following modules:

- Initializing module : describe the objects' parameter
- "Real World" simulation module: according to setting up the objects' dynamics models and calculating the models to simulate all kinds of the moving states of the robots and ball ,and send the data message to other modules
- Judgement module: according to the rule established by FIRA in world cup'98 ,applying the elicitation arithmetic to judge whether the robots foul or not reasonable.
- Strategy module: according to analyzing the data message from the "real world",work out the reasonable strategy and optimizing the track of the robot, so as to enactment the robot's wheel-rate

The whole structure is shown as figure1.

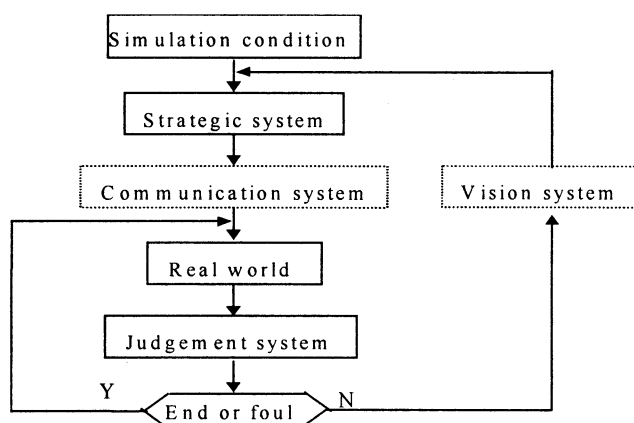


Figure 1: System structure

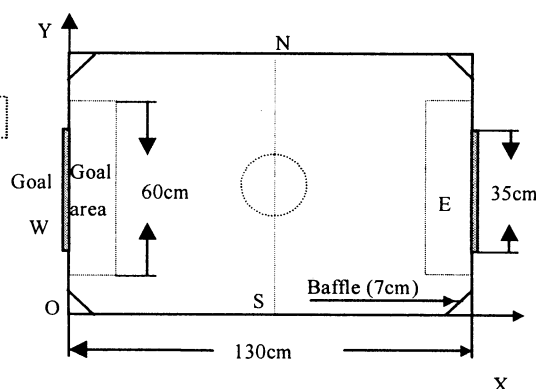


Figure 3: the planform of the ground

### 2. "Real World" simulation structure

The simulation process of the "Real World" is how to set up all the objects' models in the ground and how to solve the models. The modeling process includes setting up ground model, setting up all objects' moving states models, setting up collision models between the objects, and so on. Through analyzing the process of the real system, felling

together all the models reasonable and then applying the computer visual technique, the real system's running state can be lively presented on the computer screen. Figure 2 show the whole structure.

### 3. Ground model

The ground is shown as figure3. It can be easily known that the models of walls around the ground and all areas in the ground are simple line equation or circle equation. For instance, the equation of the left-up baffle :  $y = kx + b, k = +1, b = 83, x \in [0, 7], y \in [83, 90]$ , as well as it ,we can get the other models such as baffles, walls and goal zeros.

### 4. Objects' moving model

#### 1) Robot's moving model

Suppose a robot at a certain place  $(X_0, Y_0, \theta_0)$ , its left wheel's speed and right wheel's speed are  $V_l, V_r$ , so its moving disciplinarian as follows:

Instantaneous speed of the robot's center ( $V_c$ ):

$$V_c = (V_l + V_r) / 2$$

Orientation of the robot's instantaneous speed ( $\theta$ ):

$$\begin{cases} V_l > V_r & \theta \text{ is in same direction of } V_l \\ V_r > V_l & \theta \text{ is in same direction of } V_r \end{cases}$$

Angle speed of the robot's center ( $\omega$ ):

$$\omega = (V_l - V_r) / 7.5$$

After a time cycle ( $T$ ), the place of the robot:

$$\begin{cases} X_T = X_0 + V_c \times \cos(\theta) \times T \\ Y_T = Y_0 + V_c \times \sin(\theta) \times T \\ \theta_T = \theta_0 - \omega \times T \end{cases}$$

#### 2) Ball's moving model:

Suppose the ball at a certain place  $(X_0, Y_0)$ , its center speed is  $V_0$ , the orientation of the speed is  $\alpha_0$ , then the moving disciplinarian as follows:

The distance ( $S$ ) that the ball moves in a period

$$\text{time } (T): S = V_0 \cdot T - \frac{1}{2} \cdot a \cdot T^2, a = \mu g$$

Where  $\mu$  is the frictional coefficient between the ball and the ground,  $g$  is the gravity acceleration.

$$\text{After a period time } (T), \text{ the place of the ball: } \begin{cases} X_T = X_0 + S \cdot \cos(\alpha_0) \\ Y_T = Y_0 + S \cdot \sin(\alpha_0) \end{cases}$$

### 5. Collision model

The objects in the ground can take place all kinds of collisions, most of which are eccentricity collisions. We can model in two steps: measure the collision and describe the collision's result.

#### i) Collision between the robots

Measure condition: shown as figure 5. According to calculating the distance between the robots, when the distance from the point A to DE is less than  $\varepsilon$ , it can be believed that the two robots collided, A is the collision point.

Description the collision's result: As the collision of the two robots is eccentricity oblique collision, and the collision is stretch, we should apply the equation of the resuming coefficient relation besides the momentum theorem and momentum distance theorem, thus the moving states after the two robots collide can be gotten. (assuming the collision between the robots is lubricity)

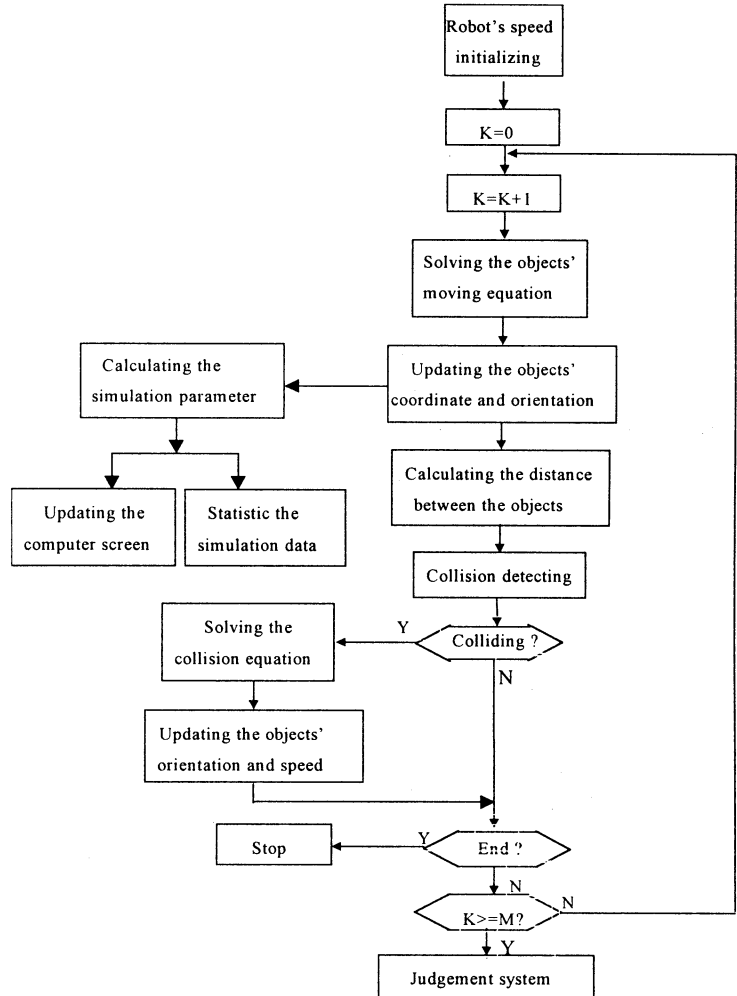


Figure 2: "Real World" structure

$$\begin{aligned}
&\text{for the I robot: } \begin{cases} M_1 \cdot U_{ct1} - M_1 \cdot V_{ct1} = 0 \dots\dots\dots(1) \\ M_1 \cdot U_{cn1} - M_1 \cdot V_{cn1} = -S \dots\dots\dots(2) \\ M_1 \cdot r_1^2 \cdot \Omega_1 - M_1 \cdot r_1^2 \omega_1 = -S \cdot C_1 B \dots\dots\dots(3) \end{cases} \\
&\text{for the II robot: } \begin{cases} M_2 \cdot U_{ct2} - M_2 \cdot V_{ct2} = 0 \dots\dots\dots(4) \\ M_2 \cdot U_{cn2} - 0 = S \dots\dots\dots(5) \\ M_2 \cdot r_2^2 \cdot \Omega_2 - M_2 \cdot r_2^2 \omega_2 = -S \cdot R \dots\dots\dots(6) \end{cases} \\
&\text{else: } \begin{cases} U_{an} = U_{cn1} + \Omega_1 \cdot C_1 B \\ V_{an} = V_{cn1} + \omega_1 \cdot C_1 B \\ U_{bn} = U_{cn2} + \Omega_2 \cdot R \\ V_{bn} = V_{cn2} + \omega_2 \cdot R \end{cases} \\
&K = \frac{U_{bn} - U_{an}}{V_{an} - V_{bn}} \dots\dots\dots(7)
\end{aligned}$$

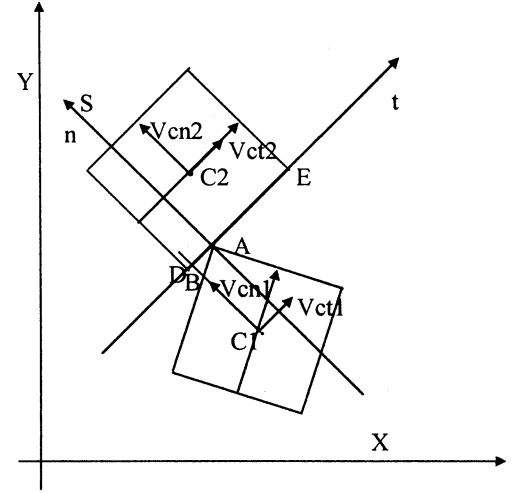


Figure 4: the collision between robots

where:  $M_1, M_2$  is the robots' mass,  $C_1, C_2$  is the robots' center of mass,  $V_{cn1}, V_{cn2}$  is the robots' speed in normal direction,  $V_{ct1}, V_{ct2}$  is the robots' speed in perpendicular normal direction,  $\omega_1, \omega_2$  is the angle speed before collision,  $r_1, r_2$  is the robots' radius of gyration,  $K$  is the resume coefficient,  $U_{cn1}, U_{cn2}$  is the robots' speed in normal direction after collision,  $U_{ct1}, U_{ct2}$  is the robots' speed in the perpendicular normal direction after collision,  $\Omega_1, \Omega_2$  is the robots' angle speed after collision,  $S$  is the impulse,  $R = \frac{1}{2} \cdot 7.5$  is the half length of the robot's size

we can obtain the follow equation:

$$\begin{bmatrix} U_{ct1} \\ U_{ct2} \\ U_{cn1} \\ U_{cn2} \\ \Omega_1 \\ \Omega_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 - \frac{K+1}{M_1 \cdot R_2} & \frac{K}{M_1 \cdot R_2} & -\frac{K+1}{M_1 \cdot R_2} \cdot CB & \frac{K+1}{M_1 \cdot R_2} \cdot R \\ 0 & 0 & \frac{K+1}{M_2 \cdot R_2} & -\frac{K}{M_2 \cdot R_2} & \frac{K+1}{M_2 \cdot R_2} \cdot CB & -\frac{K+1}{M_2 \cdot R_2} \cdot R \\ 0 & 0 & -\frac{K+1}{M_1 \cdot R_2 \cdot r_1^2} \cdot CB & \frac{K}{M_1 \cdot R_2 \cdot r_1^2} \cdot CB & 1 - \frac{K+1}{M_1 \cdot R_2 \cdot r_1^2} \cdot CB^2 & \frac{K+1}{M_1 \cdot R_2 \cdot r_1^2} \cdot CB \cdot R \\ 0 & 0 & \frac{K+1}{M_2 \cdot R_2 \cdot r_2^2} \cdot R & \frac{K}{M_2 \cdot R_2 \cdot r_2^2} \cdot R & \frac{K+1}{M_2 \cdot R_2 \cdot r_2^2} \cdot CB \cdot R & 1 - \frac{K+1}{M_2 \cdot R_2 \cdot r_2^2} \cdot R^2 \end{bmatrix} \cdot \begin{bmatrix} V_{ct1} \\ V_{ct2} \\ V_{cn1} \\ V_{cn2} \\ \omega_1 \\ \omega_2 \end{bmatrix}$$

$$\text{where: } R_2 = \frac{1}{M_1} + \frac{1}{M_2} + \frac{CB^2}{M_1 \cdot r_1^2} + \frac{R^2}{M_2 \cdot r_2^2}$$

By solving the above matrix equation, we can obtain the two robots' center speed in the normal direction ( $U_{cn1}, U_{cn2}$ ), the speed in the perpendicular normal direction ( $U_{ct1}, U_{ct2}$ ) and the angle speed ( $\Omega_1, \Omega_2$ ). The robot's wheel-speed can be gotten through these messages, which is used as the input of the robot moving model. Then the robot's moving state would be obtained.

According to the same principle, we can get the other collision models (the process is omitted).

ii) Collision between the robot and the wall

$$\begin{bmatrix} U_{ct} \\ U_{cn} \\ \Omega \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{BC^2 - r^2 \cdot K}{BC^2 + r^2 \cdot K} & 0 \\ 0 & \frac{-2 \cdot BC \cdot K}{BC^2 + r^2} & 0 \end{bmatrix} \cdot \begin{bmatrix} V_{ct} \\ V_{cn} \\ \omega \end{bmatrix}$$

iii) Collision between the ball and the wall

$$\begin{bmatrix} U_{ct} \\ U_{cn} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & K \end{bmatrix} \cdot \begin{bmatrix} V_{ct} \\ V_{cn} \end{bmatrix}$$

iv) Collision between the ball and the robot

$$\begin{bmatrix} U_{cl1} \\ U_{cl2} \\ U_{cn1} \\ U_{cn2} \\ \Omega \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -\frac{r_1 \cdot K}{r_1^2 + CB^2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & \frac{M \cdot r_1^2 \cdot K}{(r_1^2 + CB^2) \cdot M_B} & 0 & 1 \\ 0 & -\frac{K \cdot CB}{r_1^2 + CB^2} & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_{cl1} \\ V_{cl2} \\ V_{cn1} \\ V_{cn2} \\ \omega \end{bmatrix}$$

## 6. Realize the simulation system

Basing on the above system models, the paper's author develop the MIROSOT simulation system using VB5.0 for WINDOWS98. In the period of development, some questions must be thought about. Because of limitation of paper's size, we discuss the questions briefly.

### 1) Determine the simulation tempo

i) Vision sample period of time  $T_v$ : The robot soccer system is detected by the vision system. During the vision period of time, the system should complete a series of reasoning of the strategic system, A/D transition and mode identification of the vision system, and sending the command messages to robots.  $T_v$  is determined by the identifying rate of the vision system and the complexity of the strategy arithmetic, normally it is several tens milliseconds.

ii) Simulation calculating period of time  $T$ : Usually the simple time numerical value repeated method is used to solve the kinematics equations and dynamics equations, In order to improve the simulation precision, we usually chose  $T$  small enough (e.g.  $T=1\text{ms}$ ). Usually  $T$  can divides  $T_v$  exactly, such as  $M=T_v/T$ . The computer's CPU is quick enough that the system can complete all the calculation in very short time ( $\Delta t$ ), which much less than time  $T$ . Thus a 10-minute game may be completed all the simulation calculation in several seconds or even less. Because the simulation should possess the third dimension, so the delay time ( $T - \Delta t$ ) should be determined in the reality simulation course in order to make the simulation time same as the reality time.

### 2) Judgement system

By analyzing the date message from "real world", according to the game rules(3)(4), the system could take a serial of illuminated rules so as to judge all the fouls reasonably in the game, the rules is gotten in the form IF----THEN.

- IF a stalemate occurs for 10 second in the goal area THEN call a Penalty-Kick
- IF a stalemate occurs for 10 second outside the goal area THEN call a Free-Ball
- IF a goal has been scored THEN call the game commenced
- IF two robots collide acuity THEN call Free-Kick

It is not difficult to see that the judgement system is a sort of knowledge system, the rules are formed by the game's rules established by FIRA in world cup'98.

### 3) Strategic system

Since the strategic system is the soul of the whole robot soccer system, The quality of it will directly effect the result. As the real football game, all the drillmasters should think over how to attack and how to defense, so the system is different from other systems, no best, but better. Now the mainly strategy of MIROSOT are "every man for himself", "zone defense", and so on. The decision-maker from the North Eastern University team (NEW NEU) set up a six step model(2), which includes message initializing (speed calculate), attack and defense analyzing, team order determine and role assigning, target determining, track optimizing, wheel speed and displacement determining, and so on.

## 7. Conclusion

Because robot soccer system is a complex intelligent system, the development for the IROSOT simulation system includes not only a series of physical modeling and solving for the "real world", but also dealing with some knowledge question and strategy. This paper introduces a lot of exploring work about the MIROSOT simulation system modeling, which establishes a reliable foundation for the MIROSOT strategic system.

### REFERENCES

1. DaZhi Gao, New research field of the robot - robot soccer, Proc. of the robot soccer seminar'98, ShenYang, P.R.C.
2. Xinhe Xu, Six step illation model of the robot soccer, Proc. of the robot soccer seminar'98, ShenYang, P.R.C.
3. FIRA, The Laws of the Game, Proc. of the robot soccer seminar'98, ShenYang, P.R.C.
4. FIRA NEWSLETTER, Proc. of the robot soccer seminar'98, ShenYang, P.R.C.
5. FengRui, Design and Develop Simulation system of S—Mirosot, Proc. of the robot soccer seminar'98, ShenYang, P.R.C.
6. H.S.Shim, M.J.Jung, H.S.Kim, I.H.Choi, J.H.Kim Development of Vision-based Soccer for Multiagent Cooperative Systems, Proc. of Mirosot'97, Taejon, Korea, pp29~35
7. Lecture Notes on Multi-Robot Cooperative System Development(1998), Profeffor Jong-Hwan Kim, Taejon, Korea

## Role Level Design in a Hybrid Control Structure for a Vision-Based Soccer Robot System

H.-S. Shim, M.-J. Jung, H.-S. Kim, J.-H. Kim and P. Vadakkepat  
Dept. of Electrical Engineering, KAIST

Taejon-Shi, 305-701

E-mail:{hsshim, mjjung, hskim, johkim and prahlad}@vivaldi.kaist.ac.kr

### Abstract

A hybrid control structure for vision-based soccer robot system is proposed in this paper. This structure is composed of four levels namely, the role, action, behavior and execution levels. The control structure, which is a combination of hierarchical and behavioral structures, can meet the behavior and design specifications of a soccer robot system. Among the four levels, only the role level is dealt with in this paper. The role level assigns roles and areas to each robots as per the situation. The role level is required for more efficient cooperation among the agents when they have the same ability at the action, behavior and execution levels.

## 1 Introduction

The hierarchical and behavioral structures for intelligent control systems with variable, uncertain and/or complex environments, capable of achieving different aims like stability, reliability, etc are compared in [1]. It can be observed that these two structures are not suitable for a particular problem at hand.

A hybrid type control structure (Figure 1) for vision-based soccer robot system [2, 3] is composed of four levels namely, the role, action, behavior and execution levels. The control structure [4], which is a combination of hierarchical and behavioral structures, can meet the behavior and design specifications of a soccer robot system [5]. The execution level is for commanding the actuators of the robot. The behavior level is based on the behavioral structure design with basic moving behaviors, and can deal with dynamic environments (obstacle avoidance behavior). The action level is designed for playing soccer games.

The design methods for role levels, in a hybrid control structure for vision-based soccer robot system, is described in this paper.

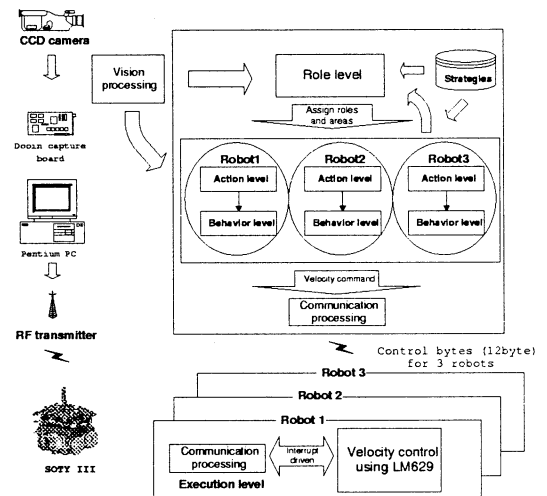


Figure 1: Control structure of SOTY III

## 2 Role Level

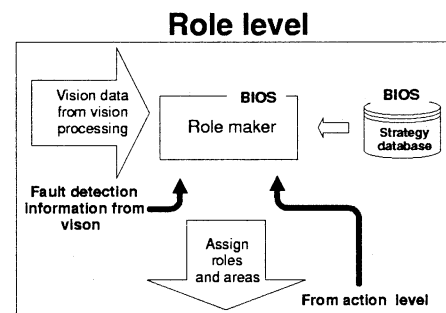


Figure 2: Role level

The role level is required for more efficient cooperation among the agents when they have the same abilities at the action, behavior and execution levels. The game can be played without the role level since the ac-

tion level has its own strategy (zone defense). However due to the problems associated with zone defense (inefficient zone assignments), robot malfunctioning (system fault state), or when opponent robots block the home robots, it is better to have the role level. The role maker in role level assigns roles and areas to each robot based on the vision information, user specified strategies, information from lower action level, etc. This is similar to the action selector in action level and the behavior selector in behavior level, in that it solves conflicts among different agents. The advantages of using the role maker are improvement of reliability, degree of fulfillment of the system goals and easiness in modeling other agents.

## 2.1 The advantage of role level

The aim of any controller designed for cooperative work is to improve the system reliability and the degree of fulfillment of system goals. The role level can provide these in soccer game. For example, in a soccer game, it is highly competitive and the environment is very dynamic. The game strategy used is an important part of the robot soccer game when two competing teams have similar abilities (robot, vision system). The reduction in competition potential due to broken down or malfunctioning robots is very severe in a game. During the game, if an attacker robot is broken down or if it is not able to carry out its actions with specified roles to each robots, it will be difficult to score a goal when a situation best for shooting turns up. If the role maker can re-assign the defender robot as an attacker robot, it is desirable under such a situation. Then, a question arises: "If no roles are assigned to the robots, then any one of them will be able to shoot the ball as an attacker. Will such a situation be better than the case where all the robots are assigned with their own roles?"

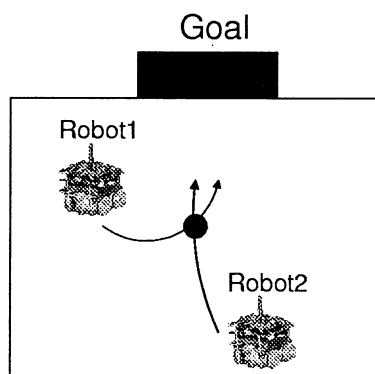


Figure 3: Who will shoot?

Consider the situation in Figure 3. Both robots are in a position to shoot, if they have the same abilities but without any specific roles assigned to them. This is a situation of conflict and they may collide if both try to shoot. An easy solution to this is to allow the robot which is close to the ball to shoot, and the other robot to take the role as a defender. However, it may not be an efficient way to select a robot close to the ball to shoot the ball, instead a robot's moving pattern is able to be looked into. For example, if robot 1 is close to the ball but its orientation and angle errors are more than those of robot 2, then the robot 1 may take longer time to shoot and it will be inefficient. It is very difficult to consider all cases like this. Another situation is to introduce learning in the system. However, learning in a complex environment, as in a robot soccer game, is quite difficult.

The role assignment of robots may be a useful method to construct a controller easily and to make them learn. If robot 2 is an attacker and robot 1 is a defender (Figure 3), robot 2 can select an action to attack and robot 1 can go for an action to defend. But, it is an inefficient role assignment when robot 1 is assigned as attacker and robot 2 as defender. In this case, the role maker would better switch their roles for better performance. Implementation of algorithms for cooperative behavior can be easily done, as the system has the role level, if the number of the situations / cases to be considered with role assignment is less than the number of those without role assignment.

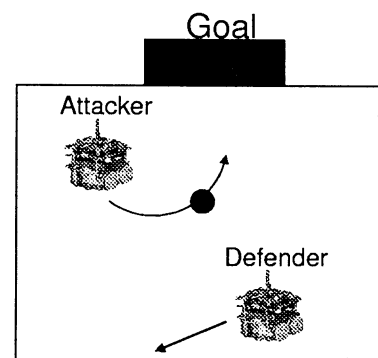


Figure 4: Inefficient role assignment

The role maker also needs to model other agents. Modeling of other agents is a very important research area in multi-agent cooperation system. The best way to achieve good cooperation among agents is to go for agent to agent communication. In a robot soccer game, if the robots can send a message like "I will shoot or I will pass the ball to you" to other robots,



the cooperation among them will be efficient. But, it is difficult to go for agent to agent communication using hardware system like the RF modem due to different technical issues. As useful communication channels are limited in number, the communication protocol may become complicated for efficient communication. On the other hand, the reliability of the communication system is very crucial in such a situation, as any malfunctioning or defect in the same will cause very severe problems in cooperative behavior. Therefore, an agent to agent communication scheme has to be dealt with carefully.

An alternative method is to make an agent to take a decision according to other agents' actions through reasoning. It can be seen in the human activities that it is possible to successfully complete tasks through cooperation even without any communication (conversation) among them. Of course, taking a decision according to other agents' actions may cause other problem like the prisoner's dilemma game, but it can be solved by processing the information intelligently. The modeling of other agents is very important in cooperative works. It is easy to understand actions or behaviors of other agents using the role assignment as an agent's action is defined as per its role.

## 2.2 Design of the role maker

The aim of any controller designed for cooperative work will be to improve reliability and the degree of fulfillment of the system goals. The role maker is constructed to accomplish the following three tasks and to assign the roles and areas of action to the three robots:

- Basic formation maintenance
- Efficient role changing
- Fault tolerance

### 2.2.1 Basic formation maintenance

There are different formation possible in a robot soccer game. Typical candidates are, '1-0-2' (one goalkeeper, 2 attackers) and '1-1-1' (one goalkeeper, one defender, one attacker). The roles and areas of action of the robots must be assigned as per the formation selected. A basic formation must assign one robot as goalkeeper because of the defense zone rule [5]. The roles and areas of the robots (other than the goalkeeper) can be selected from a set of defined roles (striker, sweeper, right\_wing, left\_wing, etc.) For instance, a '1-0-2' formation is possible with a goalkeeper

and two strikers. Another possibility is to have a goalkeeper and, one left\_wing and one right\_wing attackers. The role selection has to be done according to the game strategies. Different game strategies can emerge, based on the role and area allocation schemes, for the same formation. The area for a selected role should be assigned according to the following thumbs rules:

- The union of areas allotted to the different robots should cover the complete playground.
- There should not be any overlaps between the areas of any two robots.

### 2.2.2 Efficient role changing

In the beginning, the reliability and degree of fulfillment of system goals will be low, as the robot actions and behaviors may not be completely developed. Due to slow robots and inefficient control algorithm, the robot actions such as shoot. may not be successful. It may be desirable to change the roles (Figure 4) between a striker (right\_wing) and sweeper (left\_wing).

The robot used are good in straight motion but are not so in turning. It also takes long time to shoot when the ball is not located in front of a robot. When a robot is close to the ball and oriented towards it, efficient attacking is possible. The following switching condition is used for role changing:

It is assumed that, there are two robots, a defender and an attacker.  $d_1$  is the distance from the attacker to the ball and  $d_2$  is that from the defender to the ball.  $\theta_1$  and  $\theta_2$  are the angles between the heading direction of each robot and ball. When the following conditions are satisfied, the roles of the defender and attacker are interchanged.

$$d_2 < 2 \cdot d_1, \text{condition1 and } \theta_1 < \theta_2 \quad (1)$$

where *condition1* is satisfied when the defender robot can select a standard\_shoot action [4].

The above role changing condition is founded to be good in almost all situations. In some cases, during role interchanging, home robot was found to collide each other due to the imperfect obstacle avoidance algorithm. The aim of role changing is to improve reliability and degree of fulfillment of the system goals. It is difficult to accomplish them if the role maker is not designed carefully. In the role maker, only IF-THEN conditions were used which are not enough to face the

dynamic environment associated. For a better performance, learning ability has to be imparted to the robot soccer system [6].

### 2.2.3 Fault tolerance

When opponent robots block the home robots, or when a robot malfunctions (system fault state), it may not be possible to accomplish cooperative actions. In this case, the role level must be able to identify an agent's fault state and re-assign roles and areas to normal state robots for cooperative action among the rest of the robots. The algorithm for fault tolerance is implemented for the above mentioned two situations. The fault state of a robot can be monitored by the vision processing module. If a robot remains stand-still or if it is not able to accomplish its assigned role for a shorter duration, then the fault state is due to blocking. On the other hand, if the fault persists for a longer period, then it is inferred that the robot is malfunctioning. Such a fault can be hardware based or due to completely discharged batteries. Role and area re-assignment has to be done for the normal state robots in this situation.

In case a robot is blocked by an opponent, the roles of the striker (left\_wing) and the sweeper (right\_wing) can be interchanged as done in the efficient role changing scheme. For instance, if the right\_wing robot in shoot action is blocked by an opponent, then both robots will stand face to face and a freeball may be called. The role of the blocked right\_wing robot can be interchanged with that of the left\_wing at this instance. Such an interchange may assign any other action (for example, wandering) to the blocked robot and the position\_to\_shoot action to the left\_wing robot. This will free the blocked robot and, a possible stalemate and consequence freeball situation can be avoided. These kinds of role changing will improve system reliability. The algorithm used for role change should be designed very carefully. In case of robot malfunctioning, a simple algorithm can be used to re-assign role and areas to the normal state robots. This can be based on the basic formation maintenance scheme. If only a single robot is in normal state, it can be assigned as a goalkeeper with the complete playground as its area. In case of two normal state robots, one of them can be assigned as a goalkeeper and the other as the right\_wing attacker with the complete playground as its area.

## 3 Conclusion

A hybrid type control structure for vision-based soccer robot system is composed of four levels namely, the role, action, behavior and execution levels. Design methods of role level in a hybrid control structure for vision-based soccer robot system is described in this paper. The role level assigns roles and areas to the robots as per the situation. The role level is required for more efficient cooperation among the agents when they have the same abilities at the action, behavior and execution levels.

## References

- [1] R. E. Fayek et. al., "A System Architecture for a Mobile Robot Based on Activities and a Blackboard Control Unit," *IEEE Robotics and Auto. Conf.*, pp. 267-274, 1993.
- [2] J.-H. Kim, H.-S. Shim, H.-S. Kim, M.-J. Jung, I.-H. Choi and K.-O. Kim, "A Cooperative Multi-Agent System and Its Real Time Application To Robot Soccer," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Albuquerque, New Mexico, pp. 638-643, 1997.
- [3] H.-S. Shim, H.-S. Kim, M.-J. Jung, I.-H. Choi, J.-H. Kim and J.-O. Kim, "Designed Distributed Control Architecture for Cooperative Multi-agent System and Its Real-Time Application to Soccer Robot," *Journal of Robotics and Autonomous System*, Ed. J.-H. Kim, Vol.21, No.2, pp. 149-165, September 1997.
- [4] H.-S. Shim, M.-J. Jung, H.-S. Kim, J.-H. Kim and P. Vadakkepat, "A Hybrid Control Structure for Vision Based Soccer Robot System," *Int. Journal of Intelligent Automation and Soft Computing*, will be published in 1998.
- [5] FIRA Homepage, <http://www.fira.net>
- [6] J.-H. Kim, K.-C. Kim, D.-H. Kim, Y.-J. Kim and P. Vadakkepat, "Path Planning and Role Selection Mechanism for Soccer Robots," *Proc. of the IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgium, pp. 3216-3221, 1998.

## Keynote Speech

# Bio-informatic Coordination and Interaction among Artifacts and Humans

Osamu Katai\* and Tetsuo Sawaragi\*\*

\* Dept. of Systems Science, Graduate School of Informatics, Kyoto University  
katai@i.kyoto-u.ac.jp

\*\* Dept. of Precision Engineering, Graduate School of Engineering, Kyoto University  
sawaragi@prec.kyoto-u.ac.jp  
Sakyo-ku, Kyoto 606-8501, Japan

## Abstract

In this paper, we will present the basic concepts and a rough sketch of our project "System Theory of Bio-Informatics and its Extension towards Engineering Design Theory" in Research for the Future (RFTF) Project of Japan Society for the Promotion of Science (JSPS).

First the basic concept and our way of treating information and coordination of activities and functions in living organisms and systems are introduced, which elucidate the essential features of bio-informatics based approach to coordination, communication and interaction among agents, artifacts (modules), humans, and natural and social environments.

Particular emphases are on the complexities of relationship among humans, artifacts and environments and also the design of modules and the coordination of interaction among modules.

*Key words: Bio-informatics, design theory, coordination, interaction*

## 1 Introduction

In this paper, we will explain the basic concepts and the objectives of our project "Theory of Bio-informatics and its Extension towards Engineering Design Theory" which started in 1997 founded by JSPS in the RFTF Project.

It would be said that the basic concepts and the theories of "Bio-informatics" are still under long-term investigation, and there would be rather wide spread interpretations of bio-informatics. We, however, think that there will be certain essential features of bio-informatics that would be characterized by the rela-

tionships or interactions between information and the characteristic phenomena in living systems, i.e., the life itself.

First of all, we will explain our basic conception of this specificity by referring to the functioning of non-living systems, i.e., artifacts. Then we will focus on the objectives of our project by referring to one of our project researches. Finally, the whole scope of research interests and activities are briefly introduced.

## 2 On Artifacts-Based Interactions

As mentioned in the Introduction, artifacts are used to coordinate crude interaction between humans and physical environment which results in higher and extended interactions ranging from simple physical actions to very sophisticated communications thus realizing complex objectives via substantiating various kinds of tasks and functions.

So the artifacts may be regarded as a sophisticated and complex communication channels which relate objectives (designed goals) to physical world (field) substantiating the objectives. Fig 1 depicts this information transmission process via "coding" and "decoding" the humans' intentions[1],[2].

Concerning the question on the key issues on this information transmission process, Suh[3] introduced a quite interesting theory which any artifact should be subject to. This theory is sometimes referred to as Axiomatic Design Theory, for it provides ways of designing artifacts in a deductive manner by referring to theorems (corollaries) that are derived from just two fundamental axioms: axiom 1 (independence of functions) and axiom2 (minimization of information content).

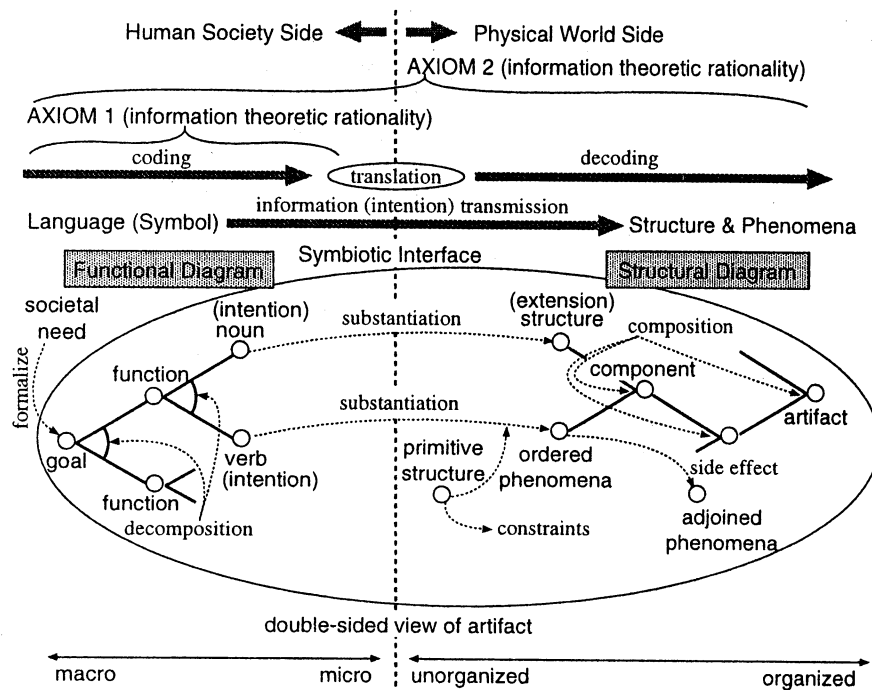


Figure 1: Information transmission model of artifacts

The first axiom searches as for the functional design of an artifact whose substantiation results no physical interactions with each other between components (elements) which substantiate (implement) different functions. So, if a physical interaction (correlation) happens to be the case, then designers should modify the design of the artifact by inserting elements which intermediate these interacting elements thus reducing or eliminating the interactions which violate axiom 1. Namely, this axiom says that you should avoid “cross talk” between the information transmission channels realizing different functions.

The second axiom eventually extends the first axiom and says that design should be minimized in its “information content”. Roughly speaking, “the simple is the best” is the guiding principle which artifacts should be subject to.

From these axioms, it can be readily seen that they regulate and coordinate the interactions between humans and physical world in such a way that the interactions become simple and “decoupled”.

Namely, independent and minimized elements (modules) realize decoupled and simple interactions that “rationally” realizes humans’ objectives.

This is quite in contrast with the specificities of interactions in biological systems, where these inter-

actions are supported by very “redundant” elements which inevitably yield complex interactions of interactions, i.e., cross talks that are prohibited by axioms 1 for rational artificial systems, i.e. artifacts.

### 3 Intelligence and Biological Interactions

The physical phenomena or interactions involved in biological systems or biological phenomena are specific in the sense that these phenomena or interactions are related in general to wide range of levels of interactions from the lower “physical level” to the higher “symbolic” or “intelligent” level.

For instance, as shown in Fig. 2, Osgood et al.[4] proposed representational mediation process model which seems to be a natural extension of behaviorism by Watson and also neo-behaviorists of Hall, Tolman and Skinner. In this model, it is explained that symbolic level information may influence several abstraction levels of interactions depending on the context of messages. This may be more directly elucidated in the well known “modern” model of Human Information Processing called SRK model[5] shown in Fig. 3. In this model, sensory inputs (sensed data) are processed

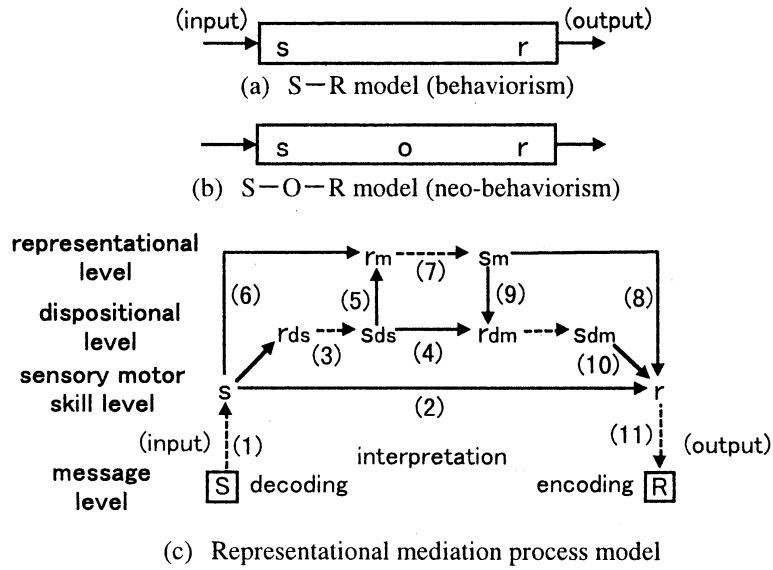


Figure 2: Human behavior models

via abstraction or association processes yielding three possible levels of interactions.

These models are restricted ones only clarifying “intelligent” human information processing. Namely, they do not model “information processing” by living systems in general. This is because biological interactions in the lower levels involve very many redundant elements.

#### 4 Symbolism and Emergent Interactions

As you may know one of the most fundamental rule of inference is *modus ponens* which is sometimes referred to as a “detachment” rule, for it cuts-off the consequent part B from an implicational from (proposition) “if A, then B” provided that A is proved to be the case. In this very highly symbolized level of information processing, to divide or to cut-off something seems to form very essential part of the processing. In other words, the intelligence at the highest (most abstract) level takes the form of dividing or cutting off something.

At the lowest level of interactions which may be seen as not being information processing nor intelligent interactions but may be seen as merely physical interactions or physical phenomena, the interactions may yield “fluctuations” due to unexpected interactions among modules or agents some of which may be environmental entities. This is because so many ele-

ments are involved in the lowest level of interactions or information processing in biological systems, and this results in generation or creation of possible courses of events occurrence, thus may be seen as information generation.

This way of arranging or sorting possible interactions between humans and environments via “one-way”, rational, hierarchical, and without cross talk inter-correlations is in quite contrast with the crude direct interactions between humans and environments.

Recent revival movements of utilizing and re-evaluating the crude and direct interactions such as “Behavior-based AI” and “Ecological Interface” [6] are very interesting to our objective of utilizing information specific in biological systems or biological phenomena in coordinating interactions among humans artifacts and environments.

In these movements (approaches), self-organizing and emerging characteristics of the fields of interactions between biological systems and environments and also the poiesis of meaning relationships among entities related the interactions play the main role in organizing and coordinating the interactions.

Utilizing artifacts through this way of developing interactions is also related to our physical existence, i.e., bodies and to proficiency of skills. Our bodies are eventually one of the essential parts of the filed of interactions, and these interactions encounter with each other in our body. Many researchers emphasized the importance of body such as “embodiment” by Brooks. Moreover, by extending a “body”

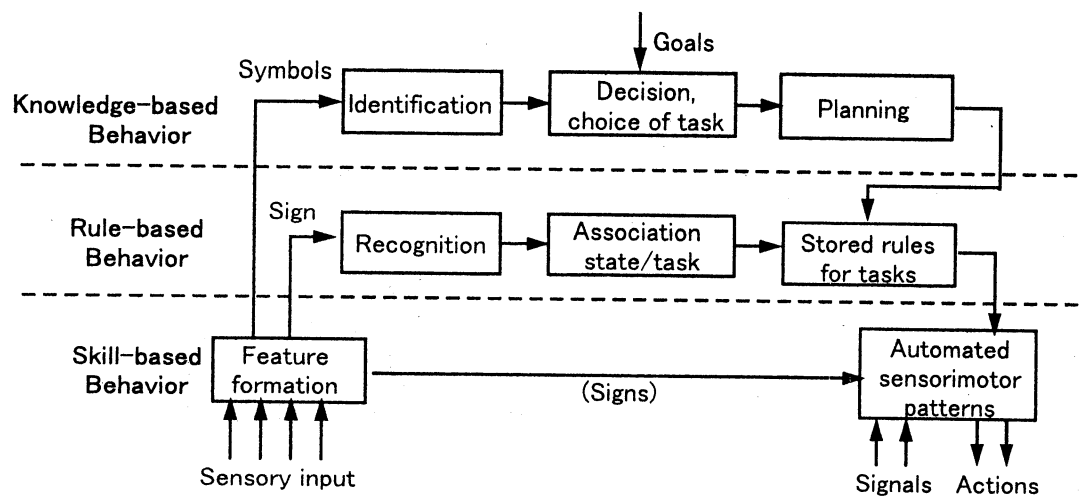


Figure 3: SRK Model of Rasmussen[5]

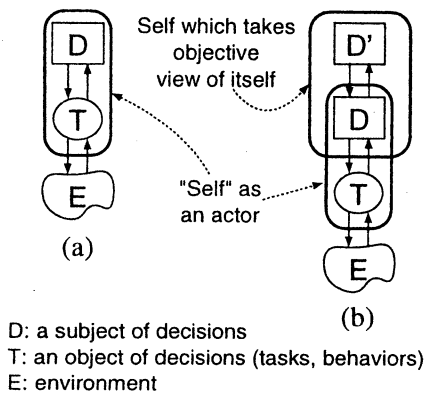


Figure 4: Two-way and emergent interactions

to more global things available to an actor such as language, communication channels, sensations, artifact tools, etc. and putting an emphasis on embedded, socially interactive domain enabled by those, Winograd and Flores[7] termed such an aspect as a "consensual domain" and a famous philosopher Merleau-Ponty called it "intercorporeite".

These "two-ways" and emergent interactions between a human body and an environment can be used to "enlarge" the "boundary" of our body by utilizing artifacts via proficiency of skills as delineated in Fig. 4.

If our tasks or objectives become complex, then the intermediating (implementing) artifacts will be complex as well. In this case as depicted in SRK model of Rasmussen in Fig. 3, the enlargement of our body is

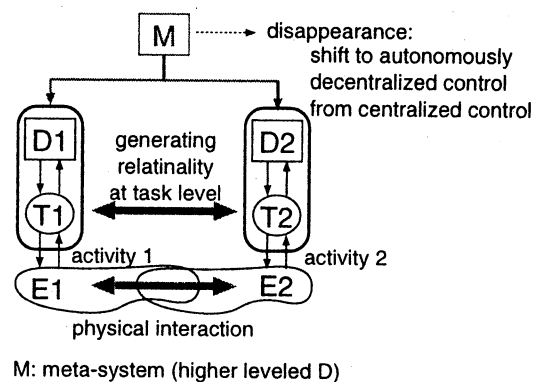


Figure 5: Self as a "meta-level"

not an easy problem. One way for solving this is to refer to Gibson's ecological psychology[8], which utilizes natural functions (values) of environments sometimes referred to as "affordances".

## 5 Creative Interactions and Human Skills

Focusing on proficient skills, this may be modelled by introducing "another" self that is "meta-level" self shown in Fig. 4(b). Namely, "duality of self" or "dual self" is important and the self at the meta level (meta-self) unconsciously guides the level of interactions quite "skillfully"[9].

To perform complex tasks in this line of skillful

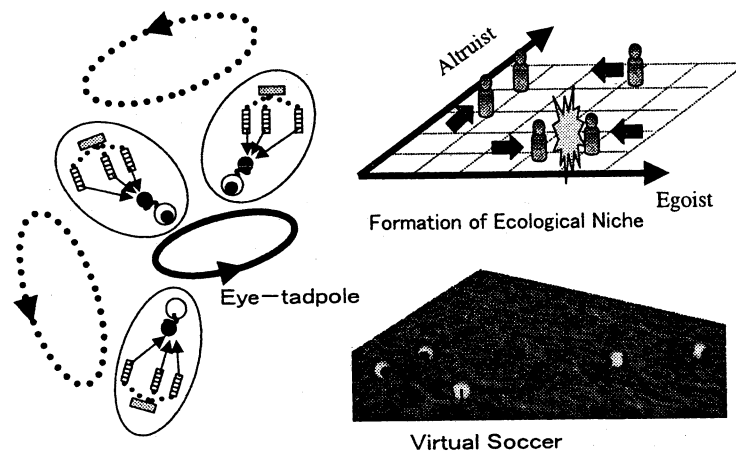


Figure 6: Generation of sociality in virtual soccer by Eye-tadpoles

behavior, we should take note of interactions among tasks. Each decision subject for each tasks is not subject to a centralized control mechanism, but controlled in an autonomously decentralized manner by creating relationality (relationships) at the task levels by referring to the real interactions in the physical environments (fields) as depicted in Fig. 5.

More radical and essential insights and the framework of the role of bio-informatics in humans-artifacts-environments interaction and coordination is proposed by Shiose et al.[10], where not only the generation of task relationality but also the tasks or the values themselves are created through the interactions in the outside physical world. This role differentiation is called “generation of sociality”. As an interesting testbed, we introduced “virtual soccer” by strange virtual creature called Eye-tadpole (cf. Fig. 6) which is a modification of Okada’s interesting creature “Talking Eye”[11] that is used to generate and develop natural conversational communication between humans and computers.

An Eye-tadpole acts on duality of self through guidance of “social affordance”. Its basic structure is based on Grefensette’s SAMUEL[12] which is a kind of CS (Classifier System) that searches for classifier rules (if-then rules) attaining high values of evaluation functions through evolutionary processes.

Fig. 7 shows the fundamental structures of the model, where self-reference (by one of the self) is used to value judgments about how it uses and evaluates its experience obtained in the past, and the social reference (by the other self) is used to design interactions for itself by considering the prospect of which it is engaged with other agents (eye-tadpoles), as well as en-

tities in the environments for the future. According to the evaluation, the CS yields formation of Ecological Niche shown in Fig. 6 resulting in individual characteristic behavior of each agent. Namely, social differentiation is attained by indirectly interacting with each other through the physical field of interactions, i.e., the ground of soccer game.

This research shows one possible way of creating novel tasks that naturally arise from the interactions among agents (modules) and environments, which is in quite contrast with the standard way of utilizing artifacts as mentioned in Section 2.

## 6 Relevant Theories on Information Interaction

In view of information transmission, these highly interactive communications (interactions) among agents necessitate new paradigm of treatment, for the standard way introduced by Shannon only treats one-way transmission without referring to the meaning of information which is crucial for emergence of order in the interacting filed. Concerning this problem, Barwise and Seligman[13] introduced a very interesting theory of information flow in distributed (decentralized) artifacts called “Channel Theory”[14]. It is based on Dretske’s idea on “Information Flow”[15] and also on the Situation Theory by Barwise and Perry[16]. Roughly speaking, the meaning of information in a decentralized artifact is localized by using Local Logics and the Shanon’s (Kolmogorov’s) idea of regarding information as a “partition” on space (state space or

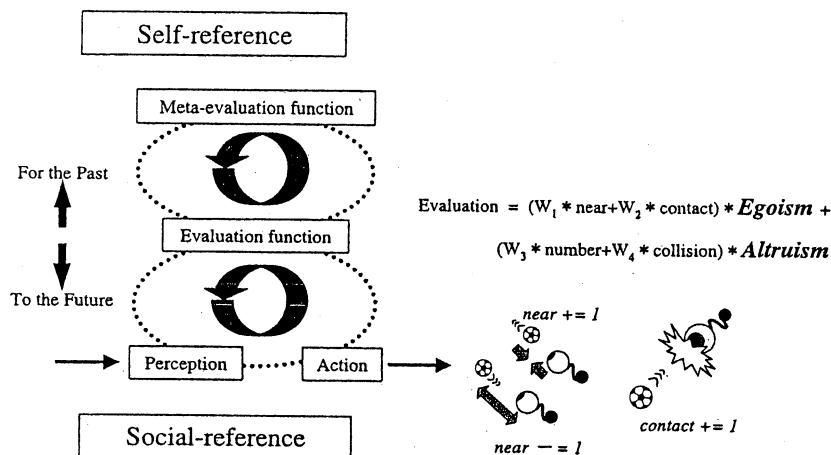


Figure 7: Bi-referential model in Eye-tadpoles

probability space) is extended and substantiated as a classification of tokens into types.

Also, it should be noted that a higher order formation in the interaction field may be better modelled by using a more abstract block diagram based on Category Theory in abstract mathematics. Rosen[17], a well-known theoretical biologist searched for the essence of life by focusing on what is the key issue which discriminate living systems from non-living machines (artifacts), and he arrived at a conclusion to use Category Theory as a representational tool for higher order relationality which characterizes biological systems.

In this sense, it is worth noting the notion of Chu Space[18], for it is closely related to the Channel Theory and also to the Category Theory. Also, Chu Space is very important for modeling concurrent and distributed processes that are essential features of the behavior of decentralized artifacts.

## 7 On Our Project on Bio-informatics

Keeping these discussions in mind, we will roughly sketch our project. The key issues we focus our attentions to are the design of modules and also the design or coordination of interactions among them.

The final objective of our research is the "total system design of large-scaled complex decentralized artifacts systems attaining *symbiosis* with humans and

natural or social environments"[19].

The basic way of approaching the final objective is to incorporate "plasticity" and "adaptivity" to the modules which will then yield a higher order self organization ability and also harmonized decentralization among modules by coordinating and organizing interactions among them.

The range of these interactions cover lower order "physical" interactions and also higher ordered "symbolized" (symbolic) interactions. We search for autonomously tuning and coordinating mechanisms of inter-level and intra-level interactions.

Thus, we finally pursuit the objective to organize "symbiotic relationships" among humans, artifacts and environments (natural or social) by augmenting the self-organizing and adaptation abilities which can cope with the complexities of physical processes and also those of human information processing.

The triad relationship among these three entities can be delineated as shown in Fig. 8. The concrete approach we adopted can be summarized in Fig. 9.

Key issues we are focusing are the module (agent) design and interaction coordination as follows:

- Modules compositions with emphases on the natural correlation and boundness among the module compositions and the coordination and organization of interactions between the modules
- The way of incorporating and implementing goal-orientedness



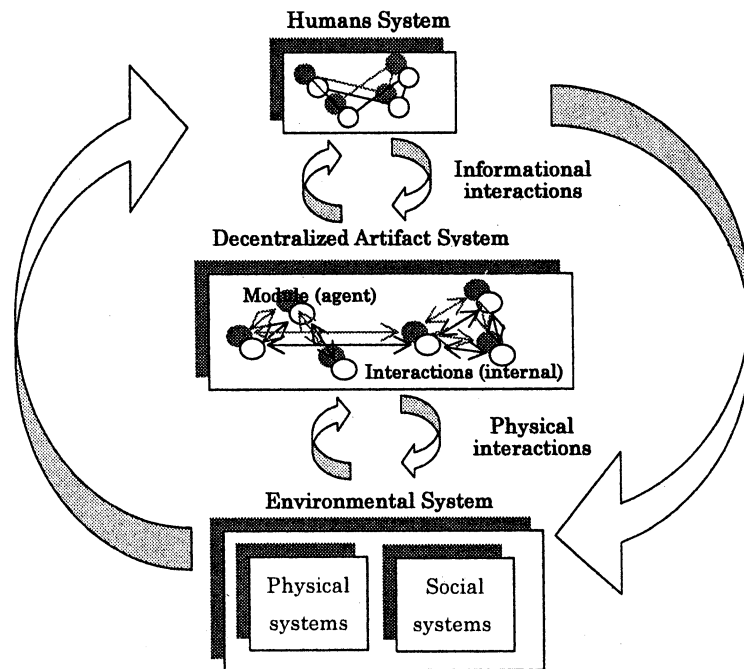


Figure 8: Interactions among Humans, Artifacts and Environments

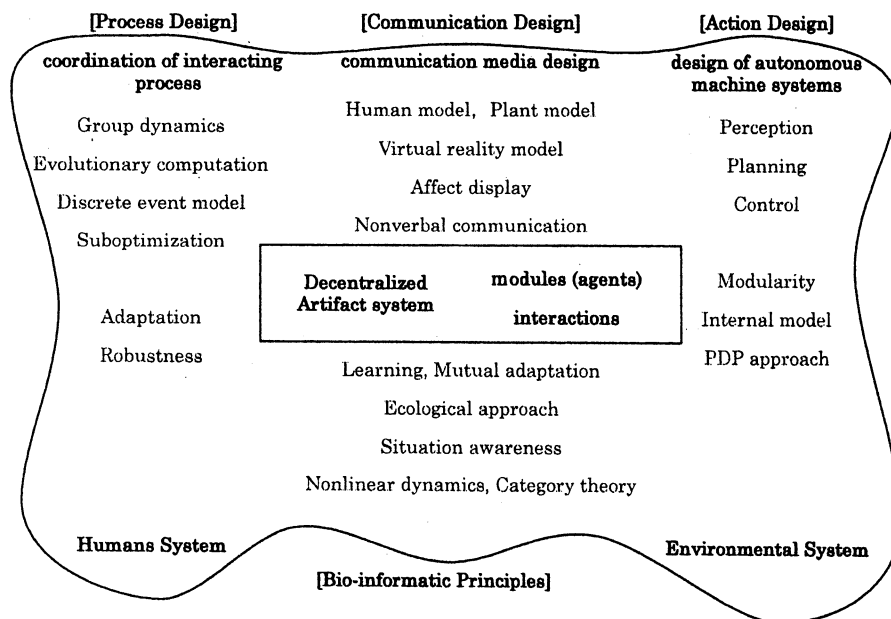


Figure 9: Activities and relationships among four Groups : Bio-informatic Principles, Action Design, Communication Design, and Process Design.

- the augmentation of the area of functioning and effects of each module
- The internalization of the complexities of outside world to the modules through the interactions with the outside world, i.e., learning, adaptation, memory, and internal modeling
- Autonomous construction of higher order relationships among modules and its destruction
- Functional differentiation of modules by self-reference and social-reference

## 8 Conclusions

We have introduced our basic idea and standpoint for treating and utilizing information and interaction in biological systems. By contrasting with the rational and efficient ways of coordinating interactions via artifacts, the specificities of biological information and the ways of utilizing them were elucidated. In order to develop more systematic and organized ways of utilizing these aspects and paradigms of bio-informatics, it is necessary and is one of our important tasks to quest for fundamental and rigorous frameworks for treating the complex and complicated interactions and flow of information among human's artifacts and environments.

## References

- [1] H. Kawakami, O. Katai, T. Sawaragi, T. Konishi and S. Iwai (1996) Knowledge acquisition method for conceptual design based on value engineering and axiomatic design theory, *Artificial Intelligence in Engineering*, Vol.10, No.3, pp.187-202
- [2] H. Kawakami, O. Katai, T. Sawaragi and T. Konishi (1997) Designing Artifacts as Symbiotic Interface between Human Society and Physical World, *Proc. of Int. Symp. on System Life*, pp.275-284
- [3] N. P. Suh (1990) *The Principles of Design*, Oxford Univ. Press.
- [4] W. Schramm (Ed.) (1963) *The Science of Human Communication*, Basic Books
- [5] J. Rasmussen (1986) *Information Processing and Human-Machine Interaction*, North-Holland
- [6] J. Flach et al. (Eds.) (1995) *Global Perspectives on the Ecology of Human-Machine Systems*, Vol.1, Lawrence Erlbaum Assoc.
- [7] T. Winograd and F. Flores (1986) *Understanding Computers and Cognition : A New Foundation for Design*, Ablex Pub. Co.
- [8] J. J. Gibson (1979) *The Ecological Approach to Visual Perception*, Houghton Mifflin Comp.
- [9] T. Sawaragi (1998) Proficient Skill Embedded with in Human, Machine and Environment, *J. of Society of Instrument and Control Engineers*, Vol.37, No.7, pp.471-476 (in Japanese)
- [10] T. Shiose, T. Sawaragi, O. Katai and M. Okada (1998) Segregated Sense of Values within Multi-agent Systems through Reciprocity Enabled by a Bi-Referential Model, *Proc. of the 2nd Japan-Australia Joint Workshop on Intelligent and Evolutionary Systems*, pp.107-114
- [11] M. Okada, N. Suzuki and S. Inokuchi (1996) Extended Chattering with Heterogeneous Minds, *Proc. of 12th Symp. on Human Interface*, pp.441-446 (in Japanese)
- [12] J. J. Grefenstette (1992) The Evolution of Strategies for Multi-agent Environments, *Adaptive Behavior*, Vol.1. No.1, pp.65-89
- [13] J. Barwise and J. Seligman (1997) *Information Flow : The Logic of Distributed Systems*, Cambridge Univ. Press
- [14] A. Shimojima (1998) What We Can Do with Channel Theory, *J. of the Japan Society for Fuzzy Theory and Systems*, Vol.10, No.5, pp.775-784 (in Japanese)
- [15] F. I. Dretske (1981) *Knowledge and the Flow of Information*, The MIT Press.
- [16] J. Barwise and J. Perry (1983) *Situation and Attitudes*, The MIT Press.
- [17] R. Rosen (1991) *Life Itself : A Comprehensive Inquiry Into the Nature, Origin and Fabrication of Life*, Columbia Univ. Press.
- [18] V. Gupta (1994) *Chu Spaces : A Model of Concurrency*, Ph. D Thesis, Comp. Sci Dept., Stanford Univ.
- [19] O. Katai (1998) Human System Interactions and Intelligent Support - A Tutorial -, *J. of Japanese Society for Artificial Intelligence*, Vol.13, No.3, pp.339-346 (in Japanese)

## A Basic Study of Virtual Collaborator - The First Prototype System Integration

H. Ishii, W. Wu, D. Li, H. Ando, H. Shimoda and H. Yoshikawa  
Graduate School of Energy Science  
Kyoto University  
Uji-shi, Kyoto, 611-0011, Japan

T. Nakagawa  
Industrial Electronics & System Development Laboratory  
Mitsubishi Electric Corporation  
Amagasaki-shi, Hyogo, 661-8661, Japan

### Abstract

The goal of this study is to develop a "virtual collaborator" as a new type of human interface environment. The virtual collaborator is an intelligent agent realized in VR space, who can communicate naturally with human like humans do with each other. As the first step of this study, the authors have constructed a prototype virtual collaborator who can behave just like plant operator in the control room of nuclear power plant, although the present prototype has no communication functions with humans. At the present stage, the virtual collaborator can detect an anomaly, diagnose the root cause and operate the control panel in accordance to the operation manual in the virtual space, where the control room of the nuclear power plant is visualized. The prototype system is constructed as a distributed simulation system, which consists of four subsystems: (1) nuclear power plant simulator, (2) man-machine interface simulator, (3) human model simulator, and (4) human body motion simulator. These subsystems have been separately developed and have been combined afterwards.

### 1 Introduction

Recently due to the increased automation by the introduction of modern computer and information technologies, the machine systems have become so large and complex that the manipulation of the machine system has become a difficult task for users. Especially, in the field of aircraft and power plants, the tendency is conspicuous and an error of an operator may give rise to serious accidents. Therefore, the study on the man-machine interface has been extensively made to

improve the relationship between human and the machine system.

In this study, the authors aim at developing a 'virtual collaborator' as an ideal human interface. The virtual collaborator is an intelligent agent robot in VR-space. Human can interact with the virtual collaborator to operate machine systems. It has a human-shaped body and can listen, talk, think and behave like humans.

In this paper, the basic concept of the virtual collaborator, the configuration of the prototype system which has been developed as the first step of this study, and the result of the experimental simulation of the system are described.

### 2 Configuration of Virtual Collaborator

In order to develop the virtual collaborator who can naturally communicate with human, the authors have contrived the configuration of the system as shown in Figure 1. The virtual collaborator has sensing system, thinking mechanism and effector system like humans. With the sensing system, the virtual collaborator can recognize the actions of human such as motion, speech, gesture and facial expression. With the thinking mechanism he can understand the situation around him and make decisions. And with the effector system, he can behave freely, speak and sometimes express his emotion to the human.

In this study, the authors have studied and developed key technologies to realize such virtual collaborator [1]. They are summarized as follows:

- Bio-Information Sensing Technology

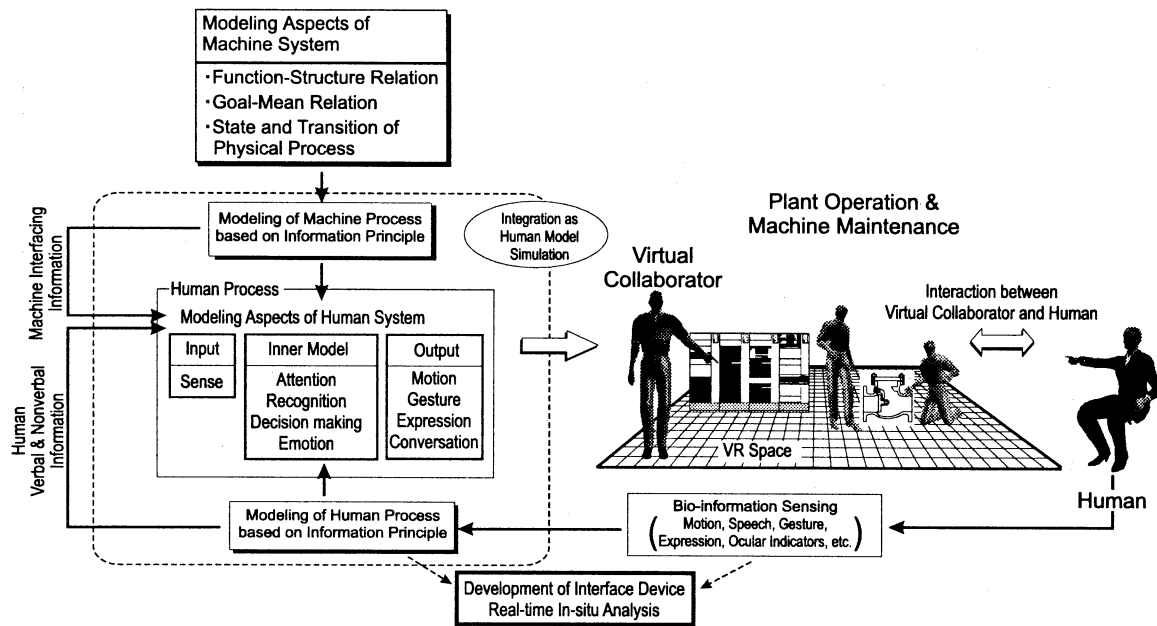


Figure 1: Overall System Configuration of Virtual Collaborator

This is the technology to estimate the human internal states such as thinking process, mental stress and emotion. In order to estimate the internal state, the targets of bio-informatic sensing are not only gesture and facial expression but also various physiological indices such as the distribution of skin temperature, the transition of the heart rate, gaze-point and the transition of the pupil size.

- Human Model Simulation Technology

This is the technology to realize the intelligence of the virtual collaborator by computer simulation. Concretely, the authors extended Reason's theory as a general human model framework. And the human model was constructed by using a development tool for real-time object-oriented expert system, G2(GenSym Co.Ltd.).

- Synthesis Technology of Human Body Motion

This is the technology to compose the body motion and facial expression as naturally as possible. In this study, in order to realize natural movement, the body motion is composed from basic motion data. In the motion database, which consists of pieces of motion data obtained by measurement of actual human motion.

In this study, as the first system integration, a prototype virtual collaborator who can behave just like

plant operator in the control room of nuclear power plant has been developed by constructing the collaborator's thinking mechanism with the Human Model Simulation Technology and the collaborator's body motion with the Synthesis Technology of Human Body Motion. At present, this prototype system has no communication function with human, but he can detect a plant anomaly, diagnose the root cause and operate the control panel in accordance with an emergency response operation manual.

### 3 System Architecture of Prototype Virtual Collaborator

In this section, the configuration of the prototype virtual collaborator is described. As shown in Figure 2, the whole prototype system is constructed as a distributed simulation system which consists of the following four subsystems: (1) nuclear power plant simulator, (2) man-machine interface simulator, (3) human model simulator, and (4) human body motion simulator. The details of the four subsystems are explained as follows.

#### (1) Nuclear Power Plant Simulator

The Nuclear Power Plant Simulator (NPP Simulator) is a real time dynamic simulator of an actual PWR plant, which can simulate various kinds

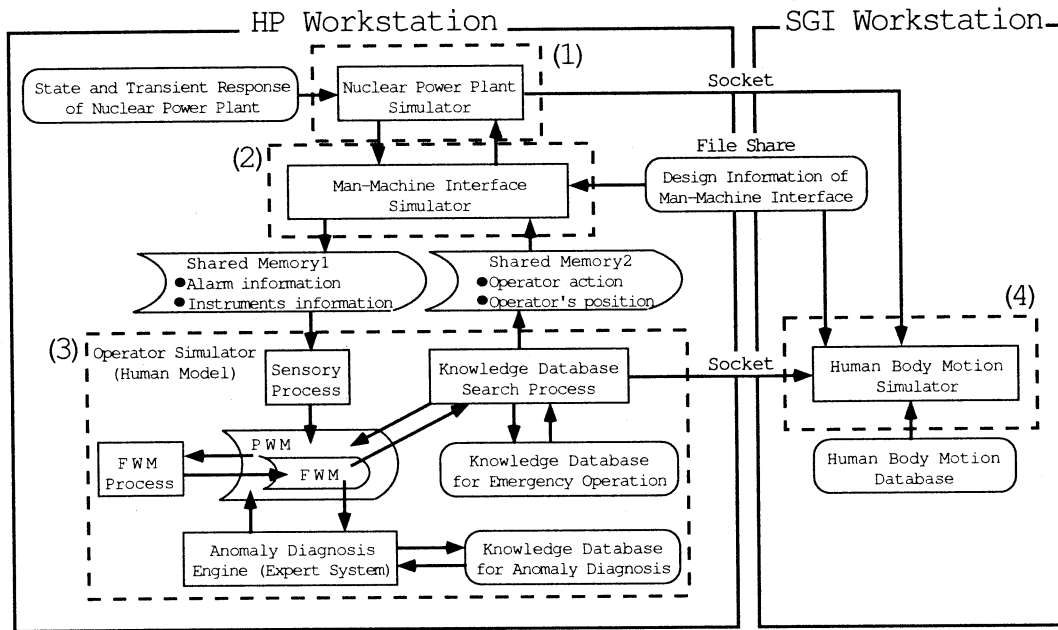


Figure 2: System Architecture of Virtual Collaborator

of plant anomalies.

#### (2) Man-Machine Interface Simulator

The Man-Machine Interface Simulator (MMI Simulator) is based on-line object-oriented database model to the presented man-machine interface information in the plant control room as 2 dimensional images. And the MMI Design Information database includes the information about layout, shape, location, panel, etc. of various equipments in the control room, and the temporal behavior of instruments.

#### (3) Human Model Simulator

The Human Model Simulator (HM Simulator) realizes 'intelligent functions' of the virtual collaborator, by employing a general human model framework which consists of sensory process, focal working memory (FWM), peripheral working memory (PWM), knowledge database, database search process, FWM process and anomaly diagnosis engine. Based on the above components, the virtual collaborator checks the control panel, detects an anomaly, diagnoses the root cause by examining hypotheses, and performs appropriate response operations. In each step, the human model simulator drives human body motion simulator in order to visualize the motion of the virtual collaborator in virtual control room. The details of

the human model simulator were described in our published paper[2].

#### (4) Human Body Motion Simulator

The Human Body Motion Simulator (HBM Simulator) generates the virtual space in real time, where not only the body motion of the virtual collaborator but also the various conditions of control room itself is visualized. In order to realize the naturalness of the body motion, first the actual human body motions of "walk" and "operation" were measured by a 3D motion capturing system, and then basic motion database was built by the measured data. When generating the body motion, the appropriate basic motion is selected from the database, and modified to fit the objective motion. By this method, the walk motion of arbitrary direction and distance, and the operation motion of pushing buttons and sliding levers can be generated naturally in real time. The details of how to measure the actual human body motion and how to compose them were described in our paper presented on EDA'98[3].

## 4 System Integration

NPP Simulator, MMI Simulator and HM Simulator are integrated on a workstation (HP VISUALIZE

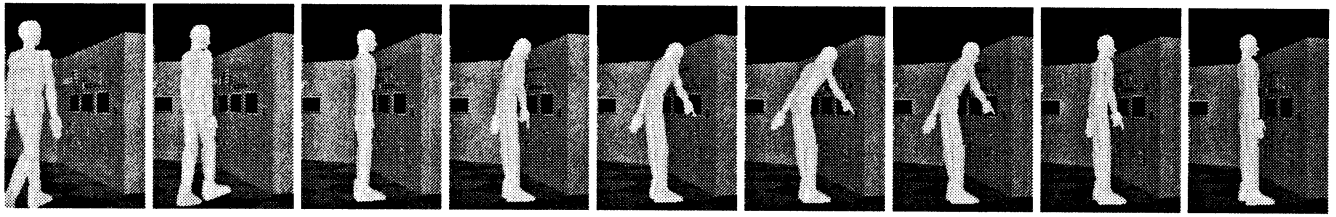


Figure 3: Composition Example of Virtual Collaborator

C200) as separate processes. And each simulator communicates with the other simulator through shared memories shown in Figure 2. The shared memory 1 deals with two types of information; (i) alarm information, and (ii) the focused MMI information. And the shared memory 2 deals with two types of information; (i) information on the operation, and (ii) MMI information, on which the operator is focusing.

On the other hand, HBM Simulator is integrated on the graphics workstation (SGI OCTANE). To visualize the virtual collaborator, it is necessary to compose not only the human body motion but also the control panel of the nuclear power plant. The design information of man-machine interface for HBM Simulator, which includes the information about the control room of nuclear power plant, is reflected to the design of the MMI Simulator.

The control room consists of 10 control panels on which about 500 instruments are located. If all the instruments are drawn with polygons, the load becomes too large, therefore the details of the instruments are drawn with textures.

The necessary information for composing the virtual collaborator in the virtual environment are (1) to which panel the collaborator moves, (2) which indicator he checks, and (3) which instrument and how he operates. In addition, the parameters of NPP Simulator must be sent to HBM Simulator to visualize the control panel. All the above information are sent by socket communication between HP workstation and SGI workstation.

By using the integrated system, some example-simulations were conducted to examine the required functions of both each subsystem and the total integrated system. As example simulations, the cases of LOCA (Leak of Coolant Accident) and SGTR (Steam Generator Tube Rupture) were simulated. Figure 3 shows an example scene in which the virtual collaborator operates a control panel in the virtual environment. From the analysis of simulation records, it was confirmed that the developed prototype system could simulate the plant operator who can detect an anomaly, diag-

nose the root cause and operate the control panel with the natural movement in virtual control room.

## 5 Conclusions

In this study, the authors have developed the prototype virtual collaborator who can operate the control panels of nuclear power plant with visualizing the control room in virtual environment by applying the technologies, which have been separately developed. The authors are planning to develop the functions to estimate human internal states and communicate with real human as the second step of this study.

## 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] H. Shimoda and H. Yoshikawa, An Experimental Study on a Virtual Collaborator as a New Human Interface (in Japanese), *The 6th Dynamics and Energy Technology Symposium*, (to be presented).
- [2] H. Yoshikawa, T. Nakagawa, Y. Nakatani, et al. (1997), Development of an Analysis Support System for Man-Machine System Design Information, *Control Engineering Practice*, Vol. 5, No. 3, pp. 417-425
- [3] H. Shimoda, H. Ando, D. Yang, et al. (1998), A Computer-Aided Sensing and Design Methodology for the Simulation of Natural Human Body Motion and Facial Expression, *Proceedings of EDA'98*, (CD-ROM)

# Generating Novel Memories by Integration of Chaotic Neural Network Modules

Akira Sano

sano@i.kyoto-u.ac.jp

Graduate School of Informatics, Kyoto University  
Sakyo-ku, Kyoto 606-8501, Japan

## Abstract

A principle of integrating neural network(NN) modules based on chaos was studied on our two-moduled Nozawa model. The chaotic NN model represents the embedded patterns as each low-dimensional periodic orbit, and the others are shown as high-dimensional chaotic orbits. It is equivalent to Freeman's "*I don't know*", and "*I know*" state. In particular, we had took notice of a condition that the two-way inputs of a NN module conflict on the embedded Hebbian correspondence. We found out that the interaction between chaotic NN modules generates the novel "*I know*" representations besides the embedded one, as novel memories.

*Keywords: Functional integration, Chaos, Neural network module, Nozawa model, Hebb rule*

## 1 Introduction

Functional localization with the module structure such as cortical column or cortical area exists on the brain. Although a large number of studies have been made on each individual function. On the other hand, these modules should functionally integrate in order that the brain works as an information system. What seems to be lacking is *how to integrate the functional modules* into a single information processing system.

Several attempts have been made to NN model the information processing by combining such multiple functional modules. These models have been applied algorithms or controllers to integrate NN modules. However, the brain autonomously integrates the modules without external controls.

We take notice of chaotic dynamics in NNs for integrating network modules. It is known that the chaotic behavior plays several important roles in artificial and real NNs. In this article, the chaotic behavior is considered from the *functional integrating* point of view.

It was found out that the connection between chaotic NN modules generated new internal states besides embedded states with Hebb rule. The existence of these new internal states suggest the novel memories or the novel functions generated by the interaction between the functional modules as a brain.

## 2 Model

### 2.1 Two-moduled Nozawa Model

A chaotic NN model is prepared to consider the interaction between the NN modules. Nozawa model [1] is one of the artificial NN models having chaotic behavior. Nozawa model is expanded from Hopfield type non-chaotic NN model [2] by adding the negative self-feedback connection at each neural element, and is proved the equivalent to Chaos Neural Network model [3].

A modular neural network model is defined based on Nozawa model to our consideration. In the following, a two-moduled Nozawa model is described by adding an external coupling term to Nozawa model [4, 8]:

$$p_i(n+1) = F_{q_i(n)}\{p_i(n)\} \quad (1)$$

$$q_i(n) = \frac{1}{T} \left\{ (1-\varepsilon) \sum_{j \neq i}^N T_{ij} p_j(n) + \varepsilon \sum_{k=1}^M T'_{ik} p'_k(n) + I_i \right\} \quad (2)$$

$$\text{with } F_q(p) = rp + (1-r) \left[ 1 - \frac{1}{2} \left\{ 1 + \tanh \left( \frac{p-q}{2\beta} \right) \right\} \right] \quad (3)$$

where  $p_i(n)$  is an internal buffer of  $i$ -th neuron at discrete time  $n$ , and  $q_i(n)$  is called control parameter of the neuron  $i$  at  $n$ .  $T_{ij}$  denotes the connecting weight

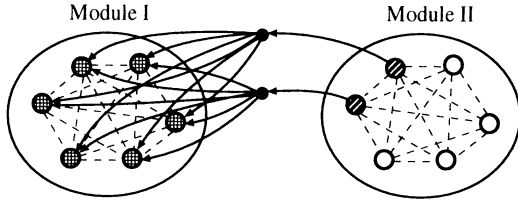


Figure 1: two-moduled (flow-type) NN model

between neurons  $j$ -th and  $i$ -th, and  $I_i$  is the threshold value of the neuron  $i$ .

In regard to eq.(2),  $p'_k(n)$  is  $k$ -th internal buffer of external module,  $T'_{ik}$  is a connection weight from  $k$ -th element of external module. The parameters  $\varepsilon$  and  $M$  denote respectively the *coupling ratio* and the *number of coupled elements* from the external module.

The constant values  $r(0 < r < 1)$  and  $\beta(> 0)$  are given as follows:

$$r = (1 - \frac{1}{R}), \quad \beta = \frac{\alpha}{RT} \quad (4)$$

where  $R$  correspond to the damping constant of Hopfield's model, and  $T$  is the connecting weight for the negative self-feedback ( $T_{ii} = -T < 0$ ). So, these constants are  $r = 0.7$ ,  $\beta = 0.006$ ,  $T = 15$  in our computational experiments below.

The conceptual diagram of two-moduled NN model is shown at Figure 1( $N = 2$ ). In this article, the inter-module coupling is restricted to one-way (flow-typed) projection as Figure 1, because of the purely observation for the influence of inter-module couplings.

## 2.2 Embedded Memories

Each module of two-moduled Nozawa model are called module I and II (see Figure 1). Now, two sets of three vector patterns are prepared for each module,  $\{C, F, 4\}$  and  $\{C', F', 4'\}$ .

The prepared sets of vector patterns are embedded into each module using Hebb rule [5]. The Hebb rule is given by the following relation, which determines the coupling weight  $T_{ij}$  between the neural elements within the modules.

$$T_{ij} = \sum_{s=1}^3 (2V_{si} - 1)(2V_{sj} - 1) \quad (5)$$

where  $\mathbf{V}_s (= \{V_{s1}, \dots, V_{s16}\}, s = 1, 2, 3)$  are embedded vector patterns.

Similarly, the coupling weight between the modules

$T'_{ik}$  are also given as follows:

$$T'_{ik} = \sum_{s=1}^3 (2V_{si} - 1)(2V'_{sk} - 1) \quad (6)$$

The vector patterns for module I and II are written as  $\mathbf{V}_s (= \{V_{s1}, \dots, V_{s16}\}, s = 1, 2, 3)$ ,  $\mathbf{V}'_s (= \{V'_{s1}, \dots, V'_{s16}\}, s = 1, 2, 3)$ , respectively. Then,  $T'_{ik}$  gives the pattern correspondences between module I and II ( $C \Leftrightarrow C'$ ;  $F \Leftrightarrow F'$ ;  $4 \Leftrightarrow 4'$ ). And, the inter-module couplings are restricted unidirectionally from module II to I.

## 2.3 External inputs

Pattern recalling characteristics of the chaotic NN model have been well known from the past few studies [6, 1, 7].

When without external input, the chaotic NN model recalls the embedded patterns wandering. And steady pattern recalling is shown with external input which has an embedded pattern. The Nozawa model has the same features of pattern recalling, too.

The external input ton the two-moduled Nozawa model is given as eq.(7) to the threshold  $I_i$  of eq.(2), according to the simple Nozawa model[1].

$$I_i = \begin{cases} 0.08 T & \text{for } k_i = 0 \\ 0.09 T & \text{for } k_i = \text{'nothing'} \\ 0.10 T & \text{for } k_i = 1 \end{cases} \quad (7)$$

The constants( $\simeq 0.09$ ) of eq.(7) indicates the control parameter  $q_i(n)$ , where we set  $T_{ij} = T'_{ik} = 0$  in eq.(7). The external input  $I_i$  is given so that  $q_i(n)$  stays near the chaotic parameter region. Let the input 16-dimensional vector pattern be  $\mathbf{K} = \{k_1, k_2, \dots, k_{16}\}$ .

## 2.4 Inconsistently condition

On the two-moduled Nozawa model, the inter-module coupling does not hinder the pattern recalling process of NN module, as long as the Hebbian correspondence is kept between each NN module[4, 8].

In this article, we would like to focus attention on the more interested condition. It is that the two-moduled Nozawa model does not have Hebbian correspondence owing to the external inputs, e.g., module I has an external input  $C$  and module II has  $F'$ . Let us call such cases *inconsistently condition* as regards embedded Hebbian correspondences. An inconsistently condition investigated in this article is shown at Figure 2.



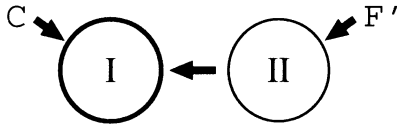


Figure 2: An inconsistently condition

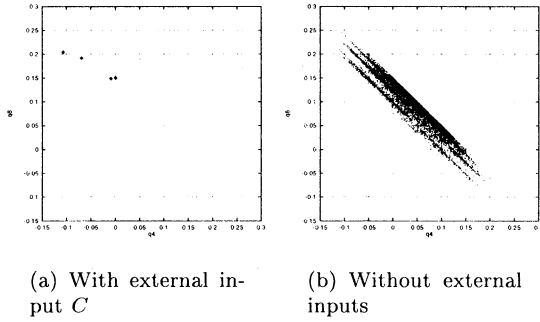


Figure 3: Internal states of the NN module( $q_4$ - $q_8$ )

### 3 Internal dynamics

#### 3.1 “I know”, “I don’t know” states

Figure 3 shows a projection with the variables  $q_4$  and  $q_8$  at module I. Internal dynamics of the chaotic NN module is broadly divided in two types. Each embedded pattern is represented as *low-dimensional periodic orbits* (Figure 3-(a)), and the others are shown as *high-dimensional chaotic orbits* (Figure 3-(b)). The periodic orbits that indicate each embedded pattern are quite unstable. It is changed to a chaotic attractor as Figure 3-(b) owing to give the external or the inter-module input in the absence of Hebbian correspondences.

The above two types internal dynamics of the chaotic NN module well agree with the role of “*novelty filter*” which is claimed based on the rabbit’s olfactory EEGs by W. Freeman. Namely, the low-dimensional attractor(Figure 3-(a)) means “*I know*” state for the embedded memories, and the chaotic attractor(Figure 3-(b)) indicates “*I don’t know*” internal state.

#### 3.2 Influence of the inter-module coupling

Also at the inconsistently condition, two-moduled Nozawa model shows “*I know*” states for the pattern  $C$

and  $F$  on the module I. However, they are destroyed in spite of with the weak inter-module coupling, because of these embedded “*I don’t know*” states are quite unstable, as we have mentioned before.

The two-moduled Nozawa model has two parameters  $M$  and  $\varepsilon$  for the strength of inter-module coupling (with eq.(2)).  $M$  is the number of inter-module coupling elements, and  $\varepsilon$  is the inter-module coupling ratio. When there is no inter-module couplings ( $M = \varepsilon = 0$ ), NN module equivalents to the (unimoduled) Nozawa model. Then, the internal dynamics of the module I obeys an external input, that is to indicate an embedded pattern  $C$ . On the other extreme case, when the inter-module coupling have most strangest parameter values ( $M = 16, \varepsilon = 0$ ), module I follows entirely the inter-module inputs from module II, that is to indicate an embedded pattern  $F$ . Both two extreme cases show the low-dimensional “*I know*” states.

However, they are destroyed in spite of with the weak inter-module coupling, because of these embedded limit cycles are quite unstable, as I have mentioned before. Hence, the inter-module couplings without Hebbian correspondences make a chaotic “*I don’t know*” state on almost all the  $M$ - $\varepsilon$  parameter space. The NN modules can not recognize these inputs which does not satisfy the Hebbian correspondences. Generally, the only internal states embedded by Hebb rule (eq.(5)) are recognized by way of the low-dimensional “*I know*” states on the chaotic NN modules, and others are said “*I don’t know*” as high-dimension chaotic attractors.

#### 3.3 Generating novel memories by inter-module couplings

However on the two-moduled Nozawa model, it is found out that the inter-module couplings do not only make chaotic “*I don’t know*” states, but also generating a few novel low-dimensional “*I know*” states. These novel “*I know*” states are observed at the inconsistently conditions.

Figure 4 shows bifurcation map of the internal buffer  $q_4$  with the parameter  $\varepsilon$ . Low-dimensional internal states (novel “*I know*” states) appear as the windows over the bifurcation structure, and the innumerable windows are shown (Figure 4). However, almost all the windows are limited to very narrow region of inter-module connection weight  $\varepsilon$ , and can not stably exist in opposition to the fluctuation of  $\varepsilon$ . At Figure 4, a low-dimensional “*I know*” state whose parameter region of  $\varepsilon$  is nearby 0.5 can only exist stably.

The low-dimensional periodic orbit which appears

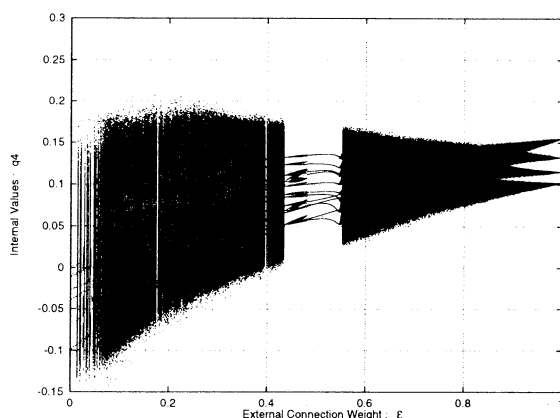


Figure 4: Bifurcation map of  $q_4$  with external connection weight  $\varepsilon$ ;  $M = 3$

over the bifurcation structure can be regarded as an novel internal representation, that means “*I know*” in the same sense to embedded patterns. However, the novel “*I know*” states do not have a correspondence to external patterns. Namely, these novel representations are generated based on only the relation between the embedded patterns, which has no Hebbian correspondences, e.g., *C-F* at inconsistently condition.

The novel “*I know*” representation that is generated by the interaction between chaotic NN modules can be considered a mechanism to process the various combinations among the NN modules, inclusive of non-experienced one. This result from the two-moduled Nozawa model suggests how the brain recognize an object that has not memorized. And then, how the brain accepts inconsistently combinations of the internal representation at each functional module can be explained.

## 4 Conclusion

The two-moduled Nozawa model has the two types internal representations, which are low-dimensional “*I know*” states and chaotic “*I don't know*” states. This categorization plays a role of novelty filter for the external inputs, like Freeman’s biological experiment. Then, we found out that the other “*I know*” states were generated by the interaction between chaotic NN modules. It can be interpreted as novel memories, furthermore as novel functions in the NNs. Our result gives an explanation how the brain autonomously integrate a lot of functional modules into an information system, and it provides a new viewpoint of chaos in artificial and real NNs.

Two-moduled Nozawa model is too abstract for a realistic model of brains, so far. However, our model is not excessively given the assumptions, thus it is sufficient simple to consider NNs generally. A multi-moduled NN model will be useful to consider the brain works, if the sources of chaotic dynamics in the brain make clear. It may give a framework to explain a multi-functional brain tasks, e.g., *Binding problem*, *Multi-modal cognition*, *Gestalt* or *Common sense*.

## Acknowledgements

I wish to thank the Japan Society for the Promotion of Science(JSPS), Research for the Future Program(RFTF), for their generous financial assistance. My special thanks are due to Professor Osamu Katai and the members of Theory Group of Bio-Informatics for valuable advices.

## References

- [1] Nozawa H (1992), A Neural Network Model as a Globally Coupled Map and Applications Based on Chaos. *Chaos*, 2(3):377–386.
- [2] Hopfield JJ (1982), Neural networks and physical systems with emergent collective computational abilities. *Proceedings of National Academy of Sciences USA*, 79:2554–2558.
- [3] Aihara K, Takabe T, Toyoda M (1990), Chaotic Neural Networks. *Phy. Lett. A*, 144:333.
- [4] Sano A (1995), The role of intersubsystem coupling density and chaotic dynamics in multi-body neural network model. In *RWC Technical Report (Proc. of the IIW95)*, RWCP, TR-95010:41–53.
- [5] Hebb DO (1949), *Organization of Behavior*. New York: John Wiley & Sons, Inc.
- [6] Adachi M, Aihara K, Kotani M (1992), Nonlinear associative dynamics in a chaotic neural network. In *Proceedings of the 2nd International Conference on Fuzzy Logic & Neural Networks*, 947–950.
- [7] Tsuda I (1992), Dynamic link of memory – chaotic memory map in nonequilibrium neural networks. *Neural Networks*, 5:313–326.
- [8] Sano A, Kunifuji S (1998), Integration of chaotic behavior modules into a flow-typed two-body neural network model. *Electronics and Communications in Japan: Part III*, 81(5):41–50.

# CONSTRAINED HIERARCHICAL PATH PLANNING OF A ROBOT BY EMPLOYING NEURAL NETS

S. Patnaik, A. Konar and A. K. Mandal

Department of Electronics and Telecommunication Engineering  
Jadavpur University, Calcutta -700 032, India

## Abstract

A new scheme for path planning of a mobile robot by artificial neural nets has been presented in this paper. The planner, has been realized by two cascaded neural nets, where the first net filters sensory information by satisfying a set of predefined constraints while the second net generates control commands for the adjustment of the velocity and directivity of the robot. The proposed system, which has been simulated on a UNIX platform, demonstrated improved performance w.r.t. the benchmarks for the navigational robots.

## 1. Introduction

Soft-computing techniques have been employed successfully in designing the planner for navigational robots[Patnaik<sup>1</sup>]. There exist quite a large number of literatures on the path planning algorithms for navigational robots. But most of these suffer from one or more of the following difficulty [Patnaik<sup>2</sup>].

- i) The criteria for minimizing energy-cost is not taken into account in the path-planning problems for existing robots.
- ii) The time constraint for reaching the goal position from a given initial position is not considered as a significant factor, though for real time systems this is an important issue.
- iii) The minimality of distance w.r.t. traversal of path are not considered in most cases.

For instance, Meng and Picton<sup>3</sup>, partitioned the entire workspace of a mobile robots into non-overlapping blocks and employed an ANN based paradigm for designing the direction of movement of the robots based on the location of obstacles in its workspace, where the minimum time and minimum distance criteria has not been considered. Michalewicz et al<sup>4</sup>, have designed an improved algorithm for path planning in a dynamic environment, avoiding collision. But the minimum energy criteria have not been taken into account. The present paper overcomes all the above limitations by employing an artificial neural net (ANN), which resembles the functional characteristic of the above three criteria concurrently.

## 2. A constraint Hierarchical scheme for path panning

A schematic overview of the two stage pipelined ANN configuration is presented in figure 1 below, where the first net filters the sensory information by using the constraints and the second net generate control commands for the movement of the motors and actuators.

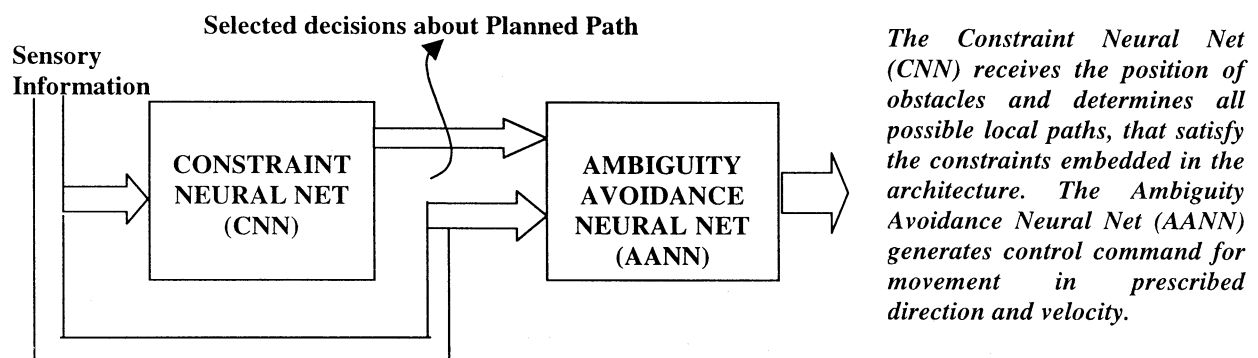


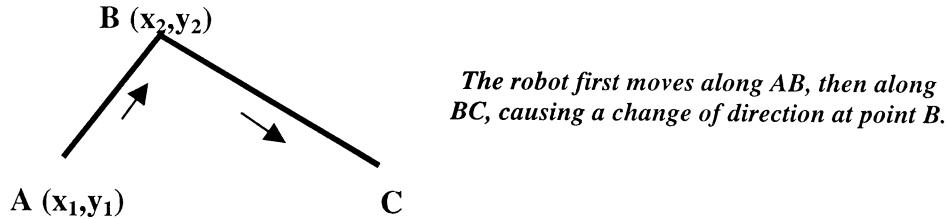
Fig.1: A schematic overview of the 2-stage pipelined ANNs for path planning and local guidance.

The CNN is designed to support the following criteria, outlined earlier:

- i) **Energy minimization Criteria:** Fewer is the change in direction / course of movement, lesser will be consumption of energy by the robot. The energy minimization criteria, formulated based on this strategy, requires movement of the robot on straight-line paths as far as practicable. This can be formally realized by minimizing,

$$\sum_{i,j} \{ (x_j - x_i)^2 + (y_j - y_i)^2 \}$$

where  $(x_j, y_j)$  is the next point following  $(x_i, y_i)$  when the robot starts changing directions. This is illustrated in fig. 2, where  $i$  is 1 and  $j$  is 2.



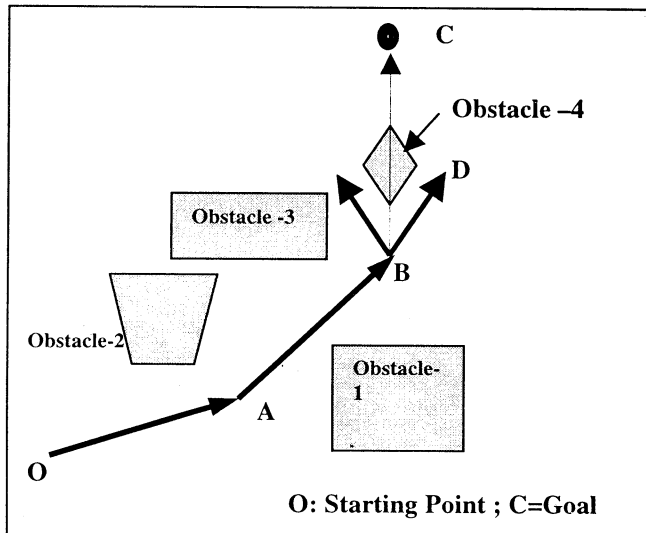
**Fig.2: The change of direction by the robot.**

- ii) **Minimality in time constraint:** This has been realized in the proposed system by keeping the speed of movement along each straight line maximized. Thus, the criteria for minimality in time constraint is satisfied if

$$\sum_{i=1}^n v_i \mid v_i = \text{velocity along the line } i.$$

$\forall i=1$   
is minimized.

- iii) **Minimality in path traversal:** This has been realized in the proposed system by moving always in the direction of goal or along the line angularly close enough to current position. Formally, the above criteria can be described (vide figure 3), as the minimization of the angle  $\theta$ , the direction of movement (say) BD, makes with the shortest route BC.



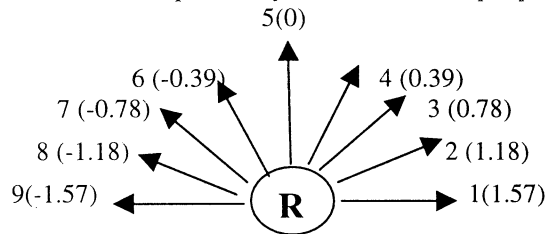
**Fig.3: Minimality in path traversal criteria.**

The criteria (i) and (ii) could be realized jointly by a single neural net. But, the output parameters for such systems being distinct, the training time will be considerably larger. So, two neural nets have been employed in the proposed system for realization of criteria (i) and (ii) separately and an AND logic following the nets determine the joint occurrences of the criteria. The direction of movement of the robot, for instance, could be identified by a multi-layered neural net. The input

and output variable for such net vide Table-I are the sensory information ( $S_i$ ) about obstacles received at different angles, while the output variables ( $d_i$ ) denotes the direction of movement of the robot. In fact with nine (9) binary input variables ( $S_1, S_2, \dots, S_9$ ) and one variable DD (Desired Direction) having a possibility space of 9, altogether we could have a possibility space of  $2^9 \times 9$  inputs. However, out of this space, less than 60 are significant from the point of view of minimum path deviation of the robot. The concept of minimum path deviation of the robot comes into picture to keep the direction of robot always aligned in the shortest route (to Goal) or angularly close to it as far as practicable.

### 3. Simulation

The speed of the robot in the direction of the movement may be estimated by a separate multi-layered neural net, whose sample entries are given vide Table-II. The input variable of this net is the measured distance of the obstacle from the robot R ( $S_i$ ), while the output variable ( $V_i$ ) denotes the velocity in the respective direction of movement of the robot. Here, nine (9) input variable ( $S_i$ ) which takes at least 11 possible decimal values (0, 0.1, ..1), could have a possibility space of  $11^9$ . However, out of this space, only few tens of the input patterns may be trained to give a generalized output



The number 1 through 9 denotes directions, where the angle between two successive directions is  $22.5^\circ$ . Direction 5(0) is the forward movement of robot. The information within parenthesis denotes the angles of the corresponding positions in radians w.r.t. the reference position 5.

Fig.4: Direction of the Robot estimated in angle (radian).

Table-I: Sample Training patterns, for the directional of movement as output and the sensory information about obstacles as input.

Input (Presence of Obstacle)										Output (Direction of Movement)								
DD	$S_1$	$S_2$	$S_3$	$S_4$	$S_5$	$S_6$	$S_7$	$S_8$	$S_9$	d1	d2	d3	d4	d5	d6	d7	d8	d9
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0.78	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0.78	0	1	1	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0

DD= Desired Direction in angle (radian);

$\forall_i S_i$ = Sensor readings, which will be 1, if the reading is 0.5 or more and 0 if the reading is less than 0.5 (where '0' means obstacle free & '1' means occupied by obstacles)

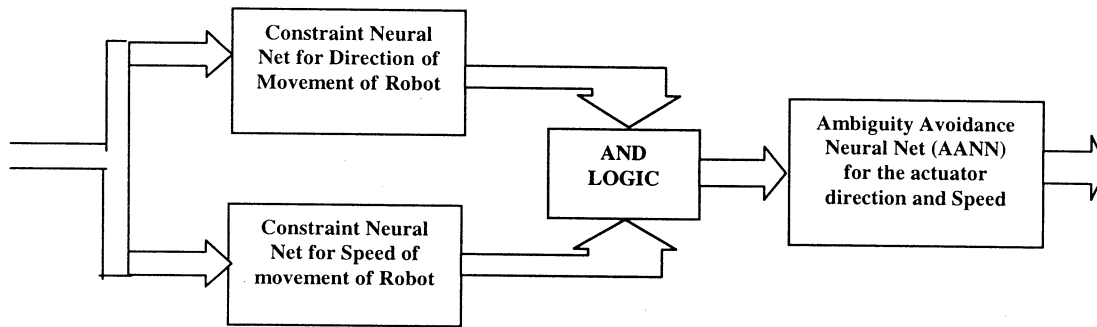
$\forall_i d_i$ = Direction of movement shown in figure 4;

Table-II: Sample Training patterns, for velocity of movement as output and the distance of obstacle form the robot as input

Input (Measured Distance from Robot)									Output (Velocity of Movement)								
$S_1$	$S_2$	$S_3$	$S_4$	$S_5$	$S_6$	$S_7$	$S_8$	$S_9$	v1	v2	v3	v4	v5	v6	v7	v8	v9
0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
0	0	0.5	0.7	1	0.6	0.8	0	0	1	1	0.5	0.3	0	0.4	0.2	0	0
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
0.1	0.5	0.7	0.9	1	0.8	0.7	0.5	0.2	0.9	0.5	0.3	0.1	0	0.2	0.3	0.5	0.8
0	0.2	0.5	0	1	0.6	0.2	0.7	0.8	1	0.8	0.5	1	0	0.4	0.8	0.3	0.2

$\forall_i S_i = (2 - d_i) / (2 - 0.2)$ , where  $d_i$  is the distance of the of the obstacle from the robot.

$Mv_i$  = Speed in respective direction as shown in the figure 4.



**Fig.5 : Outputs of the two Neural nets (Direction and Speed) are ANDed together and fed to AANN**

**Table-III: Training Pattern of Ambiguity Avoidance Neural Net (AANN) for actuator velocity and direction.**

Input (Direction and Speed of the robot in different direction)										Output	
Dl	vd1	vd2	vd3	Vd4	Vd5	Vd6	Vd7	Vd8	Vd9	V	D
0	0	0	0	0	1	0	0	0	0	1	0
0.39	0	0	0	0.9	0.9	0	0	0	0	0.9	0.39
.	.	.	.	.	.	.	.	.	.	.	.
.	.	.	..	.	0	..	.	..	.	.	.
1.18	0	0	0.8	0	0	0	0.8	0	0	0.8	0.78
-1.18	0	0	0.8	0	0	0	0.8	0	0	0.8	-0.78

Dl = Immediate past direction of movement of the robot

$\forall_i vd_i$  = ANDed result of outputs of Table-I and Table-II.

V= Final speed of movement and D= Final direction of movement

## 4. Conclusions

It may be noted that the third criteria for movement of the robot on straight line path, is satisfied automatically due to physical characteristic of direction finding along the straight line. So, no extra effort is required to satisfy it. The first and the second constraint satisfying neural nets work together in parallel and their outputs are ANDed and fed to the AANN (vide fig.5), which identifies final direction and velocity of movement, considering the immediate past direction of movement of the robot. The input-output relationship, based on which the AANN was trained is presented in table III.

The training time for the nets on SGI OCTANE machine under IRIX 6.4 operating system depends on the error margin and with a figure of 0.2%, the training time is negligibly small in the order of less than 1.5 sec.

## References

- [1] S.Patnaik, A.Konar, A.K.Mandal,(1998) ,“Navigational Planning with Dynamic Scenes”, Proc. Of the Int. Conf. On Computers and devices for Communication(CODEC-98), pp.40-43.
- [2] S.Patnaik, A.Konar, A.K.Mandal ,(1998), “Visual Perception for Navigational Planning and Co-ordination of Mobile Robots” Indian Journal of Engineers , March-April Vol., modified version of which has been communicated to IEEE Expert.
- [3] H.Meng and P.D.Picton,( 1994.) “Neural Network for Local Guidance of Mobile Robots”, Proc. Of ICARCV , Singapore,
- [4] J. Xiao, Z. Michalewicz, L. Zhang and K. Trojanowski, (1987), “Adaptive Evolutionary Planner/Navigator for Mobile Robots”, IEEE Trans on Evolutionary Computation, April , Vol.1 , No.1,

# A New Learning Method Using Prior Information of Neural Networks

Baiquan Lu, Kotaro Hirasawa, Junichi Murata and Jinglu Hu

Department of Electrical and Electronic Systems Engineering  
Graduate School of Information Science and Electrical Engineering  
Kyushu University, Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan  
Tel: (092) 642-3959 Fax: (092) 642-3962  
E-mail: lu@terra.ees.kyushu-u.ac.jp

## abstract

In this paper, we present a new learning method using prior information for three-layer neural networks. Usually when neural networks are used for identification of systems, all of their weights are trained independently, without considering their inter-relation of weights values. Thus the training results are not usually good. The reason for this is that each parameter has its influence on others during the learning. To overcome this problem, first, we give exact mathematical equation that describes the relation between weight values given a set of data conveying prior information. Then we present a new learning method that trains the part of the weights and calculates the others by using these exact mathematical equations. This method often keeps a priori given mathematical structure exactly during the learning, in other words, training is done so that the network follows predetermined trajectory. Numerical computer simulation results are provided to support the present approaches.

**Keywords:** prior information, neural network learning, part parameter learning, exact mathematical structure

## 1. Introduction

Recently, new neural networks using prior information have been reported (see [1]-[4]). In these literatures, many approaches have been proposed to incorporate prior information, which can be roughly grouped into two kinds. The first one is the methods to incorporate prior information in construction of neural network. Bayesian neural network is one of the typical examples. Another typical example is fuzzy neural network. In such approaches, it is difficult to incorporate prior information that is described by a set of explicit mathematical equations. The second one is the methods where prior information of neural networks are considered as a restriction that is usually added to the objective function [6]. However, this kind of method does not make clear the relation between prior information and neural network weights. In this paper, we present a new method for three-layered neural network

learning which utilizes the prior information expressed by a set of explicit mathematical equations and also makes clear the relation between the prior information and the weights of neural networks.

## 2. Relation between Parameters of Neural Networks and Teaching Function

In this section, we will study relation among the weights in two cases. One case is that partial derivative of teaching function  $f(x_1, x_2, \dots, x_n)$  for a variate  $x_i, i = 1, 2, \dots, n$  is known, the other is to the contrary, the derivative is not known. In the first case, we study the relation between the weights from the first layer to the second layer and the weights from the second layer to the third layer and the relation among the weights from the first layer to the second layer. In the second case, we give relation of weights by using neural network outputs at some special points.

As any mapping of  $n$ -input-one-output can be approximated by a neural network with three layers [5] for given teaching function  $f(x_1, x_2, \dots, x_n)$ , there is an integer  $m$  which satisfies the follows approximation equation with very small approximate error.

$$f(x_1, x_2, \dots, x_n) = \sum_{k=1}^m \alpha_k \varphi(W^T X + \theta_k) \quad (1)$$

where  $\alpha_k$  and  $W^T$  are weights from the second layer to the third layer and weights from the first layer to the second layer respectively,  $X = (x_1, x_2, x_3, \dots, x_n) \in R^n$  is input vector of the neural networks. Let  $\theta_k = 1$ ,  $\varphi$  is node function such as sigmoid function. We assume  $f(x_1, x_2, \dots, x_n) \in C^m$ , and let coefficients of each term of Taylor expansion of both sides of (1) equal at  $x_1 = x_2, \dots, x_n = 0$ , we have

$$\begin{aligned} f(0, 0, \dots, 0) &= \sum_{k=1}^m \alpha_k \varphi(1) \\ f_{x_{i1}}^{(j)}(0, 0, \dots, 0) &= \sum_{k=1}^m \alpha_k W_{i1,k}^j \varphi^{(j)}(1) \end{aligned}$$

$$f_{x_{i1}, x_{i2}}^{(j+1)}(0, 0, \dots, 0) = \sum_{k=1}^m \alpha_k W_{i2,k}^j W_{i1,k} \varphi^{(j+1)}(1) \quad (2)$$

where  $f_{x_{i1}, x_{i2}}^{(j+1)}(0, 0, \dots, 0)$  is the  $j_{th}$  order partial derivative with respect to  $x_{i2}$  after the first partial derivative with respect to  $x_{i1}$ ,  $j=0,1,2,\dots$ ,  $i=1,2,\dots$ ,  $x_i$  is the  $i_{th}$  input of neural network, and  $m$  is the number of hidden layer in the neural network.

Equation (2) can be written by matrix as follows

$$A_1 B_1 = C_1 \quad (3)$$

where

$$A_1 = \begin{bmatrix} 1 & 1 & \dots & 1 \\ w_{i1,1} & w_{i1,2} & \dots & w_{i1,m1} \\ \vdots & \vdots & \dots & \vdots \\ w_{i1,1}^{m1} & w_{i1,2}^{m1} & \dots & w_{i1,m1}^{m1} \end{bmatrix}, B_1 = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{m1} \end{bmatrix},$$

$$C_1 = \begin{bmatrix} \frac{f(0,0,\dots,0)}{\varphi(1)} - \sum_{k=m1+1}^m \alpha_k \\ \frac{f_{x_{i1}}^{(1)}(0,0,0,\dots,0)}{\varphi^{(1)}(1)} - \sum_{k=m1+1}^m \alpha_k w_{i1,k} \\ \vdots \\ \frac{f_{x_{i1}}^{(m1)}(0,0,0,\dots,0)}{\varphi^{(m1)}(1)} - \sum_{k=m1+1}^m \alpha_k w_{i1,k}^{m1} \end{bmatrix},$$

$$A_2 B_2 = C_2, \quad (4)$$

where

$$A_2 = \begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_{m1} \\ \alpha_1 w_{i2,1} & \alpha_2 w_{i2,2} & \dots & \alpha_{m1} w_{i2,m1} \\ \vdots & \vdots & \dots & \vdots \\ \alpha_1 w_{i2,1}^{m1} & \alpha_2 w_{i2,2}^{m1} & \dots & \alpha_{m1} w_{i2,m1}^{m1} \end{bmatrix},$$

$$B_2 = [w_{i1,1}, w_{i1,2}, \dots, w_{i1,m1}]^T,$$

$$C_2 = \begin{bmatrix} \frac{f_{x_{i1}}^{(1)}(0,0,\dots,0)}{\varphi(1)} - \sum_{k=m1+1}^m \alpha_k w_{i1,k} \\ \frac{f_{x_{i1}}^{(2)}(0,0,\dots,0)}{\varphi^{(2)}(1)} - \sum_{k=m1+1}^m \alpha_k w_{i1,k} w_{i2,k} \\ \vdots \\ \frac{f_{x_{i1}}^{(m1+1)}(0,0,\dots,0)}{\varphi^{(m1+1)}(1)} - \sum_{k=m1+1}^m \alpha_k w_{i1,k} w_{i2,k}^{m1} \end{bmatrix},$$

where  $m1 \leq m$ ,  $i1=1,2,\dots,n$ ,  $i2=1,2,3,\dots,n$ . For clarity, we assume  $m1=m$  in the following. Since the inverse of matrix  $A1$  exists in (3) if and only if (5) is satisfied for  $i1_{th}$  input of neural network, otherwise we can decrease  $m1$  until the inverse of matrix  $A1$  exists. Here we assume that inversion of matrix  $A1$  exists for  $m1=m$ , then we have (6).

$$w_{i1,k1} \neq w_{i1,k2}, \quad (5)$$

where  $k1 \neq k2$ ,  $k1=1,2,\dots,m$ ,  $k2=1,2,\dots,m$ .

$$B_1 = A_1^{-1} C_1 \quad (6)$$

From (6), we can know that  $B_1$  can be expressed by  $w_{i1,k1}$ ,  $k1=1,2,\dots,m$ , in other words, weights from the second layer to the third layer can be expressed by weights from the first layer to second layer. This is import for learning, we can use this for help learning of networks.

We have studied that relation between weights from the second layer to the third layer and weights from the first layer to second layer as mentioned above, now we study a relation of among weights from the first layer to the second layer.

If (5) and (8) are satisfied for  $i_{th}$  input, then we get (8) from (4)

$$\alpha_k \neq 0, k=1,2,3,\dots,m. \quad (7)$$

$$B_2 = A_2^{-1} C_2 \quad (8)$$

From (8), we get that  $w_{i1,k1}$  can be expressed by  $w_{i2,k2}$  for  $i1 \neq i2$ , for given  $\alpha_i$ ,  $i=1,2,\dots,m$ , under the condition of (5) and (8).  $\alpha_i$  for  $i=1,2,\dots,m$ , is not usually equal to zero, otherwise  $i_{th}$  unit of hidden layer can be removed.

So far, we discussed relation among weights under the condition that the finite derivative of  $f(x_1, x_2, \dots, x_n)$  for  $x_i, i=1,2,\dots,n$  is known. Now we discuss the case where the finite derivative of  $f(x_1, x_2, \dots, x_n)$  for  $x_i, i=1,2,\dots,n$ , is not known. Suppose we have some cardinal points of  $X$  where the network output should be exactly equal to the value of the teaching function  $f(x_1, x_2, \dots, x_n)$ . Such cardinal points can be origin, points that give maximal values, or points corresponding to the minimal value of  $f(x_1, x_2, \dots, x_n)$ . Our requirement on one of these points  $(x_1, x_2, \dots, x_n)$ , is written as

$$f(x_1(i), x_2(i), \dots, x_n(i)) = \sum_{k=1}^m \alpha_k \varphi(W^T X(i) + \theta_k) \quad (9)$$

This is another expression of prior information.

From (9) or (3) and (4), we know that if all weights learn independently by descent method, then (9) or (3) and (4) are hardly to be guaranteed at every learning step, therefore learning result is not often good. In the following section, we present a new learning method that always keeps prior exact mathematical structure, such as (9) or (3) and (4), in other words, parts of parameters are trained by descent methods, and the others are calculated to give the required trajectories. for example, (9) or (3) and (4).

### 3. Learning Using Prior Information

In this section, we will discuss how to use the prior information for learning of neural networks.



We assume that the objective function be defined as the following

$$E(W, \alpha) = \frac{1}{2} \sum_{k=1}^P [y_k - f(x_1(k), x_2(k), \dots, x_n(k))]^2, \quad (10)$$

where  $y_k$  and  $f(x_1(k), x_2(k), \dots, x_n(k))$  are  $k_{th}$  desired outputs and  $k_{th}$  actual output respectively,  $P$  is total number of train data,  $W$  a vector constituting of the weight from first layer to the second layer,  $\alpha$  a vector constituting of weights from the second layer to the third layer.

Now first, we study and give learning method incorporating the prior constraints (3) and (4), then give learning method for (9). We know, since we assumed  $m_1=m$ , that the right sides of equations (3) and (4) are constant with no relation to parameters of neural networks for a given function. If we differentiate and linearize the both sides of (3) and (4), then under the condition of (5) and (8), we have

$$\Delta \alpha = T^{(i)} \Delta W_i, \quad (11)$$

$$\Delta W_{i1} = Q^{(i1, i2)} \Delta W_{i2}, \quad (12)$$

where

$$\Delta \alpha = [\Delta \alpha_1, \dots, \Delta \alpha_m]^T, \Delta W_i = [\Delta w_{i1}, \dots, \Delta w_{i,m}]^T,$$

$$T^{(i)} = -(A1)^{-1}$$

$$\begin{bmatrix} 0 & 0 & \dots & 0 \\ \alpha_1 w_{i1} & \alpha_2 w_{i2} & \dots & \alpha_m w_{i,m} \\ \vdots & \vdots & \dots & \vdots \\ m\alpha_1 w_{i1}^{m-1} & m\alpha_2 w_{i2}^{m-1} & \dots & m\alpha_m w_{i,m}^{m-1} \end{bmatrix},$$

$$Q^{(i1, i2)} = -Q1^{(i2)}(Q2^{(i1, i2)}T^{(i2)} + Q3^{(i1, i2)}),$$

$$Q1^{(i2)} = \begin{bmatrix} \alpha_1 & \alpha_2 & \dots & \alpha_m \\ \alpha_1 w_{i2,1} & \alpha_2 w_{i2,2} & \dots & \alpha_m w_{i2,m} \\ \vdots & \vdots & \dots & \vdots \\ \alpha_1 w_{i2,1}^m & \alpha_2 w_{i2,2}^m & \dots & \alpha_m w_{i2,m}^m \end{bmatrix}^{-1},$$

$$Q2^{(i1, i2)} = \begin{bmatrix} w_{i1,1} & w_{i1,2} & \dots & w_{i1,m} \\ w_{i1,1}w_{i2,1} & w_{i1,2}w_{i2,2} & \dots & w_{i1,m}w_{i2,m} \\ \vdots & \vdots & \dots & \vdots \\ w_{i1,1}w_{i2,1}^m & w_{i1,2}w_{i2,2}^m & \dots & w_{i1,m}w_{i2,m}^m \end{bmatrix},$$

$$Q3^{(i1, i2)} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 1 & 1 & \dots & 1 \\ \vdots & \vdots & \dots & \vdots \\ mw_{i2,1}^{m-1} & mw_{i2,2}^{m-1} & \dots & mw_{i2,m}^{m-1} \end{bmatrix}$$

$$\begin{bmatrix} \alpha_1 w_{i1,1} & 0 & \dots & 0 & 0 \\ 0 & \alpha_2 w_{i1,2} & 0 & \dots & 0 \\ \vdots & \vdots & \dots & \dots & \vdots \\ 0 & \dots & \dots & 0 & \alpha_m w_{i1,m} \end{bmatrix},$$

where  $i1 \neq i2$ .  $\Delta \alpha$  and  $\Delta W_i$  are small.

By Taylor expansion at  $W$  and  $\alpha$ , we have

$$\begin{aligned} \Delta E(W, \alpha) &= \frac{\partial E}{\partial W} \Delta W + \frac{\partial E}{\partial \alpha} \Delta \alpha + \dots \\ &= \sum_{i=1}^m \sum_{k=1}^n \frac{\partial E}{\partial W_{i,k}} \Delta W_{i,k} + \sum_{k=1}^n \frac{\partial E}{\partial \alpha_k} \Delta \alpha_k + \dots \end{aligned} \quad (13)$$

where  $m, n$  are the number of input units, the number of hidden units respectively. Furthermore if (5) is satisfied for  $i2_{th}$  input unit, then we have

$$\begin{aligned} \Delta E(W, \alpha) &= \sum_{i \neq i2}^m \sum_{k=1}^n \frac{\partial E}{\partial W_{i,k}} \left( \sum_{k1}^n Q_{k,k1}^{(i, i2)} \Delta W_{i2, k1} \right) + \\ &\sum_{k=1}^n \frac{\partial E}{\partial \alpha_k} \left( \sum_{k1}^n T_{k,k1}^{(i2)} \Delta W_{i2, k1} \right) + \sum_{k1=1}^n \frac{\partial E}{\partial W_{i2, k1}} \Delta W_{i2, k1} + \dots \\ &= \sum_{k1}^n \left( \sum_{i \neq i2}^m \sum_{k=1}^n \frac{\partial E}{\partial W_{i,k}} Q_{k,k1}^{(i, i2)} + \sum_{k=1}^n \frac{\partial E}{\partial \alpha_k} T_{k,k1}^{(i2)} + \frac{\partial E}{\partial W_{i2, k1}} \right) \\ &\Delta W_{i2, k1} + \dots \end{aligned} \quad (14)$$

According to the gradient descending method, we get a learning algorithm as follows

$$\begin{aligned} \Delta w_{i2, k1} &= \\ -\eta \left( \sum_{i \neq i2}^m \sum_{k=1}^n \frac{\partial E}{\partial W_{i,k}} Q_{k,k1}^{(i, i2)} + \sum_{k=1}^n \frac{\partial E}{\partial \alpha_k} T_{k,k1}^{(i2)} + \frac{\partial E}{\partial W_{i2, k1}} \right), \end{aligned} \quad (15)$$

where  $k1 = 1, 2, \dots, m, \eta$  is the learning rate,  $i2$  is the number of inputs which satisfy (5). Weights can be adjusted by (11), (12) and (15). First, we get  $\Delta w_{i2, k1}$  by using (15), next get  $\Delta w_{i1, k1}$  by using (12), then get  $\Delta \alpha$  by using (11).

Now we discuss a learning method using (9) with given  $L$  cardinal point data (without loss of generality, we let  $L \leq m$ ), we have

$$\begin{aligned} f(x_1(1), x_2(1), \dots, x_n(1)) &= \sum_{k=1}^m \alpha_k \varphi(1) \\ &\vdots \\ f(x_1(L), x_2(L), \dots, x_n(L)) &= \sum_{k=1}^m \alpha_k \varphi(L) \end{aligned} \quad (16)$$

where  $\varphi(i) = \varphi(W^T X(i) + 1)$ . If we differentiate and linearize the both sides of (16), then we have

$$\begin{aligned} \sum_{k=1}^m \varphi(1) \Delta \alpha_k + \sum_{k=1, j=1}^{m, n} \alpha_k \dot{\varphi}(1) x_j \Delta w_{j,k} &= 0 \\ &\vdots \\ \sum_{k=1}^m \varphi(L) \Delta \alpha_k + \sum_{k=1, j=1}^{m, n} \alpha_k \dot{\varphi}(L) x_j \Delta w_{j,k} &= 0 \end{aligned} \quad (17)$$

where  $\Delta w_{j,k}, \Delta \alpha_k, k = 1, 2, \dots, m, j = 1, 2, \dots, n$  are small,  $\dot{\varphi}$  is the first order derivative of  $\varphi$ .

Among  $\Delta w_{j,k}$  and  $\Delta \alpha_k$ ,  $L$  parameters are chosen so that an inverse of their coefficient matrix exists and these parameters can be expressed by the others in (17). If the inverse matrix is not exist, then we decrease  $L$  until the inverse matrix exists. For simplicity, we assume that chosen  $L$  parameters are  $\alpha_k, k = 1, 2, \dots, L$ , then we have

$$AB = -CD, B = FD, F = -A^{-1}C, \quad (18)$$

where

$$A = \begin{bmatrix} \varphi(1) & \varphi(1) & \cdots & \varphi(1) \\ \varphi(2) & \varphi(2) & \cdots & \varphi(2) \\ \vdots & \vdots & \cdots & \vdots \\ \varphi(L) & \varphi(L) & \cdots & \varphi(L) \end{bmatrix}, \quad B = \begin{bmatrix} \Delta \alpha_1 \\ \Delta \alpha_2 \\ \vdots \\ \Delta \alpha_L \end{bmatrix},$$

$$C =$$

$$\begin{bmatrix} \varphi(1) & \cdots & \varphi(1) & \dot{\varphi}(1)x_1(1) & \cdots & \dot{\varphi}(1)x_n(1) \\ \varphi(2) & \cdots & \varphi(2) & \dot{\varphi}(2)x_1(2) & \cdots & \dot{\varphi}(2)x_n(2) \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \varphi(L) & \cdots & \varphi(L) & \dot{\varphi}(L)x_1(L) & \cdots & \dot{\varphi}(L)x_n(L) \end{bmatrix},$$

$$D = [\Delta \alpha_{L+1}, \Delta \alpha_{L+2}, \dots, \Delta \alpha_m, \Delta w_{1,1}, \dots, \Delta w_{n,m}]^T,$$

From (13), we can get (19) by (18).

$$\Delta E(W, \alpha) =$$

$$\begin{aligned} & \sum_{i=1, i_2}^{n, m} \left( \frac{\partial E}{\partial w_{i_1, i_2}} + \sum_{k=1}^L \frac{\partial E}{\partial \alpha_k} F_{k, (i_1 * m + i_2 + m - l)} \right) \Delta w_{i_1, i_2} \\ & + \sum_{j=L+1}^m \left( \frac{\partial E}{\partial \alpha_j} + \sum_{k=1}^L \frac{\partial E}{\partial \alpha_k} F_{k, (j-L)} \right) \Delta \alpha_j, \dots \end{aligned} \quad (19)$$

According to the gradient descending method, we get the following algorithm for learning.

$$\begin{aligned} \Delta w_{i_1, i_2} &= -\eta \left( \frac{\partial E}{\partial w_{i_1, i_2}} + \sum_{k=1}^L \frac{\partial E}{\partial \alpha_k} F_{k, (i_1 * m + i_2 + m - l)} \right) \\ \Delta \alpha_j &= -\eta \left( \frac{\partial E}{\partial \alpha_j} + \sum_{k=1}^L \frac{\partial E}{\partial \alpha_k} F_{k, (j-L)} \right), j = L+1, \dots, m, \end{aligned} \quad (20)$$

where  $\eta$  is the learning rate. Weights of neural networks except for  $\Delta \alpha_j, j = 1, 2, \dots, L$ , can be adjusted by (20), while  $\alpha_j, j = 1, 2, \dots, L$  can be adjusted by (18).

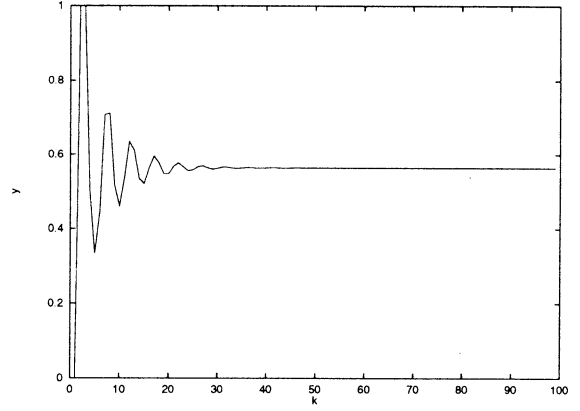


Figure 1: a plot of simulation function

In the above, we have got learning method by using prior information. This means that part of parameters are adjusted by the learning method which uses local information, other part of parameters are adjusted by exact calculation which uses global information. A procedure of parameter adjustment based on the above method is the following:

- 1) choose  $L$  or  $m_1$  parameters among weights of neural networks so that they satisfies (5) or (18) if possible, otherwise decrease  $L$  or  $m_1$  and again choose the parameters. Weights of neural networks are initialized by random, except for parameters number  $L$  or  $m_1$ , since parameters number  $L$  or  $m_1$  can be obtained from (16) or (3) and (4).
- 2) part of parameters (except for parameters number  $L$  or  $m_1$ ) are adjusted by (15) or (20), other part of parameters are adjusted by (18) or (11) and (12).
- 3) choose  $L$  or  $m_1$  parameters among weights of neural networks so that they satisfies (5) or (18) if possible, otherwise decrease  $L$  or  $m_1$  and again choose the parameters, then go to step 2), until error function is changed to satisfactory value.

#### 4. Simulation

In this section, the simulations are carried out for identification of a dynamic plant in order to verify the above idea. In this example, the plant is assumed to take the form:  $y_p(k+1) = P[y_p(k), y_p(k-1), y_p(k-2), u(k), u(k-1)]$  where the unknown function  $P$  has the form:

$$P[x_1, x_2, x_3, x_4, x_5] = \frac{x_1 * x_2 * x_3 * x_5(x_3 - 1) + x_4}{1 + x_3^2 + x_2^2}, \quad (21)$$

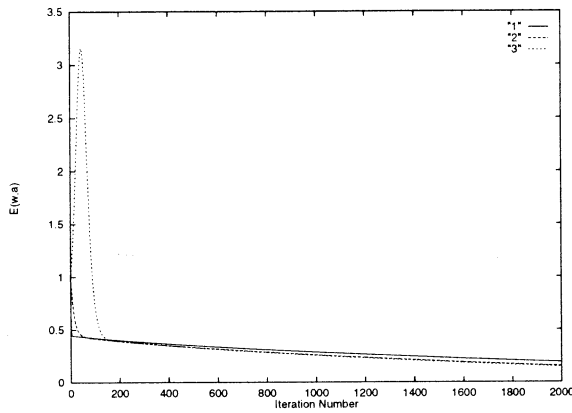


Figure 2: plot 1,2,3 are conventional EBP where learning rate is 0.0012,0.0015,0.00155 respectively

In the condition of  $u(k) = 1, k > 0$ , the plot of  $y_p(k+1)$  is shown in Fig.1, where the number of samples is 100. Neural network is formed as 5-60-1, node function is  $f(x) = (1 - e^{-x})/(1 + e^{-x})$ , the error function is (10). Because points of  $k=1,2,5,7,10,12,80$  is used as special points of the plot,  $L=7$ . initial value of weights from the first layer to the second layer and from second to third layer except for  $\Delta\alpha_i (i = i_1, i_2, i_3, i_4, i_5, i_6, i_7)$  are generated from the uniform distribution (0.5,0) and (0.2,0) respectively, and  $\Delta\alpha_i (i = i_1, i_2, i_3, i_4, i_5, i_6, i_7)$  are calculated by(18). In this simulation,  $i$  of  $\Delta\alpha_i$  is generated from the uniform distribution (1,60) in every step in order to make (18) holds. We take 3 different values of learning rates  $\eta = 0.00001, 0.00002, 0.00005$  and compare with error backpropagation (EBP) algorithm with learning rates  $\eta = 0.0012, 0.0015, 0.00155$ . The plots of error function are shown in Fig.2 and Fig.3. We are easily known from plots that learning of neural network is much faster with a small identification error and jumps down sometimes. This is produced by  $\Delta\alpha_i (i \leq L)$ , The results of simulation indicate that proposed method is very useful.

## 5. Conclusion

In this paper, a set of formulas have been introduced for the relation of weights of neural networks and learning method which is done according to given learning trajectories using prior information of neural networks with three layers. The proposed method can make each other's influence of weights as small as possible and shows good performance in the training. It has the following characteristics : prior information is effectively used and is much faster than the conventional EBP method. Its effectiveness and applicability have been demonstrated by computer simulations.

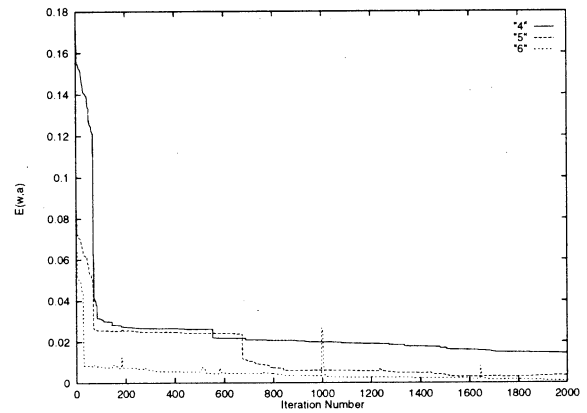


Figure 3: plot 4,5,6 are proposed method where learning rate is 0.00001,0.00002,0.00005 respectively

## References

- [1] Hyun Myung and Jong-Hwan Kim(1997), Time-Varying Two-phase Optimization and Its Application to Neural-networks Learning *IEEE Trans. on Neural Networks*, Vol.8,No.6, 1997.
- [2] Daniel T and Jenq-Neng Hwang(1997), Solving Inverse Problem by Bayesian Neural Network Iterative Inversion with Ground Truth Incorporation, *IEEE Trans. on Signal Processing*, Vol.45,No.11,1997.
- [3] Yan-Qing Zhang and Abraham Kandel(1998),Compensatory Neurofuzzy Systems with Fast Learning Algorithms, *IEEE Trans. on Neural Networks*, Vol.9,No.1,pp83-105 1998.
- [4] P.J.Angeline,G.M.Sauders and J.B.Pollack(1994), An evolutionary algorithm that constructs recurrent neural networks, *IEEE Trans. on Neural Networks*, Vol.5,pp54-65, 1994.
- [5] Tainping Chen and Hong Chen(1995), Approximation Capability to Functions of Several Variables, Nonlinear Functional and Operators by Radial Basis Function Neural Networks, *IEEE Trans. on Neural Networks*, Vol.6,No.4,1995
- [6] Chia-yiu Maa and Michael A.Shanblatt(1992),A Tow-Phase optimization Neural Network *IEEE Trans. on Neural Networks*, Vol.3,No.6,pp.1003-1009, 1992.

## Design of Neural Network Controller using Feedback Structure

Wee-Jae, Shin, Sang-yun, Lee

Dept. of Electronic Eng. Faculty of Electrical and Electronic Eng. Kyungnam University  
449 Wolyoung, Masan, Kyungnam KOREA e-mail: [cwee@hanma.kyungnam.ac.kr](mailto:cwee@hanma.kyungnam.ac.kr)

Kotaro Hirasawa

Dept. of Electrical and Electronic Systems Eng. Graduate School of Information Science and  
Electrical Eng. Kyushu University.6-10-1 Hakozaiki, Higashi-ku, Fukuoka 812-8581 JAPAN  
e-mail : [hirasawa@ees.kyushu-u.ac.jp](mailto:hirasawa@ees.kyushu-u.ac.jp)

### Abstract

As too small size of hidden layer can not propagate the learning data to model. In this paper, hidden layer of proposed controller used small units through a feedback configuration different from many cases in order to decrease hidden layer and number of unit. It was stable in systematic as lacking of learning is compensated by feedback. We were confirmed by simulation that designed recurrent NN controller have converged at a steady state without oscillation at a transient state.

### 1. Introduction

Although recurrent neural network has been possible to a application in a time and a space by the extension of learning rule of multiple layer NNs in existing. There is problem of the configuration that is demanded learning time of large numbers according to add a feedback. To solve this problem, many kind of methods related with learning of recurrent NNs were presented as follows. Pearlmutter was obtained by means of integral type the weight vector and dependence degree of time constant.

Fang presented more improved learning method that it can change the learning ratio to Delta-bar-delta rule. On the other hand, there is researching part in different side independently. It used William's equation to extend in learning of weight vector the concept of EKF. Though this method can make a fast speed of learning compare with method of Williams, there is a defect demanded much memory capacity for storage a many kind of matrix equation.

Thus many improved model using EKF was studied. Chen was studied that the size of NNs have optimal capacity according to a result of learning. Similar study was implemented by Rao also, Li was configured

The recurrent NNs of same structure through many of combination.

### 2. Recurrent Neural Network

Structure of recurrent neural network is added a feedback in multi-layer neural network of a existing which used a component of input adding by

feedback a part of output layer. These neural networks can implement effectively the dynamic characteristic of system make a feedback repeatedly.

At first, basic model is follows as

$$\tau_x \frac{dx_i}{dt} = -x_i + \sum_j \omega_{ij} f(x_j) + I_i \quad (1)$$

$\omega_{ij}$  is weight vector,  $\tau_x$  is time constant,  $f$  denote a output characteristic of neuron, equation (1) vary dynamically by a weight vector and time constant if a input is given in initial condition and it denote mathematically a motion reached gradually to saddle point and there is analogous point with dynamic differential equation of Hopfield Network make an exception of time constant structure. But it is different a point of apply a supervised learning and a point of use to the demanded object make a variation oneself of transient response. We shall denote Jacobian matrix equation to  $L_{ij}$  and then

$$L_{ij} = \delta_{ij} - \omega_{ij} f'(x_j) \quad (2)$$

We shall define the error vector  $y^0$  by equation(2)

$$y^0 = (L^T)^{-1} J \quad (3)$$

in above equation,  $x_i$  is each component of state vector  $x$ ,  $I_i$  is a external input like a bias input, gradient of weight vector  $\omega_{ij}$  is follows as

$$\nabla_{\omega} E = y^0 f(x^0)^T \quad (4)$$

where  $\delta_{ij}$  is unity matrix,  $f'(x_j)$  is derivative of  $f(x_j)$ ,  $L^T$  is transpose of  $L$  matrix equation, and  $J$  denote a error signal of external,  $x^0$  is saddle point for differential equation of neural network.

Namely, we made continuously a learning with the weight by ideal gradient then output will converge the saddle point. William and Nielsen were arranged in order to apply the spatiotemporal pattern, We shall use equation of Nielsen. Error function  $F(\omega)$  for learning defined as

$$F(\omega) = E \left[ \left\{ \frac{1}{\sum_{k=1}^{k_{stop}} U(k)} \sum_{k=1}^{k_{stop}} \sum_{n \in U(k)} [y_n(k) - \dot{y}_n(k)]^2 \right\} \right] \quad (5)$$

Where,  $U(k)$  set is consist integers of superimpose set for units given desire output on decided time and  $k_{stop}$  mean the numbers of repetition learning of input and output described  $U(k)$ ,  $E$  is desire value,  $y_n(k)$  and  $\dot{y}_n(k)$  is a desire output and actual output of  $n$ th unit in time  $k$ . a gradient of actual output  $\gamma_{nij}$  by each of weight vector  $\omega_{ij}$  is defined,

$$\gamma_{nij}(k) = \frac{\partial \dot{y}_n(k)}{\partial \omega_{ij}} \quad (6)$$

We can be derived the repetition formula

$$\begin{aligned} \gamma_{nij}(k) = & f'_n \{I_n(k)\} \{\delta_{in} z_j(k-1)\} \\ & + \sum_{p=1}^N \left\{ \omega_{n(p+m)} \gamma_{pij}(k-1) \right\} \end{aligned} \quad (7)$$

$\delta_{in}$  is kronecker delta,  $I_n(k)$  is input come to output unit of  $n$ th, in above equation  $f'_n$  is differential of output unit function of  $n$ th and through this repetition formula, the gradient of error function by means of weight vector  $\partial F(\omega)/\partial \omega_{ij}$  can be describe as equation (8)

$$\frac{\partial F(\omega)}{\partial \omega_{ij}} = E \left[ \left\{ \frac{1}{\sum_{k=1}^{k_{stop}} U(k)} \sum_{k=1}^{k_{stop}} \sum_{n \in U(k)} 2 \{ y_n(k) - \dot{y}_n(k) \} \gamma_{nij}(k) \right\} \right] \quad (8)$$

Arrange to matrix equation then a final learning rule of recurrent neural network can be described as Equation (9)

$$\omega^{new} = \omega^{old} - \alpha \nabla_{\omega} F(\omega) \quad (9)$$

Where,  $\omega^{old}$  is a weight vector before updated,  $\omega^{new}$  is a weight vector updated by learning,  $\alpha$  denote ratio of learning. Basic concept of the theory mentioned in the foregoing lies in BPTT(Back-propagation through time) Bengio presented for condition of weight vector under noise can maintain the converge value when recurrent unit converged to the any output value which associated with this recurrent neural network and Jin was survey with the dynamic characteristic and stability for a Recurrent Neural Network consist of multiple layer. Many kind of methods related with learning of recurrent NNs were presented as follows Pearlmutter was defined to network equation.

$$T_i \frac{dy_i}{dk} = -y_i + \sigma(x_i) + I_i, x_i = \sum_j \omega_{ij} y_j \quad (10)$$

Auxiliary variable  $z_i$  denote

$$\frac{dz_i}{dk} = \frac{1}{T_i} Z_i - e_i - \sum_j \frac{1}{T_j} \omega_{ij} \sigma'(x_j) z_j, z_i(k_1) = 0 \quad (11)$$

$$e_i = \frac{\partial E}{\partial y_i} \quad (12)$$

We can solve weight vector and dependence degree of time constant for index of performance given integral of division  $(k_0, k_1)$ .

$$\frac{\partial E}{\partial \omega_{ij}} = \frac{1}{T_j} \int_{k_0}^{k_1} y_i \sigma'(x_j) z_j dk \quad (13)$$

$$\frac{\partial E}{\partial T_i} = -\frac{1}{T_i} \int_{k_0}^{k_1} z_i \frac{dy_i}{dk} dk \quad (14)$$

Fang presented more improved learning method which change the learning ratio to Delta-bar-Delta rule follows as

$$\Delta \omega_i(k) = \omega_i(k) - \omega_i(k-1) = -\varepsilon_i(k) \delta_i(k) \quad (15)$$

We controlled appropriately the learning ratio as rules.

$$\Delta \varepsilon_i(t) = \begin{cases} k_i, & \text{if } \bar{\delta}_i(k-1) \delta_i(k) > 0 \\ -\Phi_i \varepsilon_i(k) & \text{if } \bar{\delta}_i(k-1) \delta_i(k) < 0 \\ 0, & \text{otherwise} \end{cases} \quad (17)$$

Where,

$$\delta_i(t) = \frac{\partial E(k)}{\partial \omega_i(k)}, \bar{\delta}_i(k) = (1 - \theta_i) \delta_i(k) + \theta_i \bar{\delta}_i(k-1) \quad (18)$$

$\theta_i$  is inertia terms, case of applying in exclusive OR problem makes control properly the direction of  $\Delta \omega_i$ , learning time is reduced 10% of Pearlmutter's method.

On the other hand, there is studying independently in different side, Williams used in learning of weight vector the concept of EKF (Extended Kalman Filter). We can describe the dynamic equation of recurrent neural network as follows,

$$y(k+1) = f(y(k), \omega(k), u(k)) \quad (19)$$

state transition matrix equation can be obtained.

$$F(n-1) = \frac{\partial y_{n-1}}{\partial x} |_{x=\hat{x}(n-1|n-1)} \quad (20)$$

$$P(n|n-1) = F(n-1)P(n-1|n-1)F(n-1)^T + Q(n-1) \quad (21)$$

If output  $Z(k)$  possible to measure then measuring matrix equation can be linearized.

$$H(n) = \frac{\partial y_n}{\partial x} |_{x=\hat{x}(n|n-1)} \quad (22)$$

We can calculate kalman gain as follow

$$K(n) = P(n|n-1)H(n)^T [H(n)P(n|n-1)H(n)^T + R(n)]^{-1} \quad (23)$$

We can rewritten the measuring value.

$$\hat{x}(n|n) = \hat{x}(n|n-1) + K(n)[z(n) - h_n(\hat{x}(n|n-1))] \quad (24)$$

$$P(n|n) = P(n|n-1) + K(n)H(n)P(n|n-1) \quad (25)$$

If output  $z(k)$  can not measure then it make a update by means of this rules.

$$\hat{x}(n|n) = \hat{x}(n|n-1) \quad (26)$$

$$P(n|n) = P(n|n-1) \quad (27)$$

Though this method can make a speed of learning compare with method of Williams, there is a defect demanded much memory for many kind of matrix equations. Thus, many kind of improved model with EKF was studied. Chen was studied that the size of NNs have optimal capacity according to a result of learning. Similar study was implemented by Rao also, Li was configured the recurrent NNs of same structure through many of

combination. But too small number of hidden layer can not propagate the learning data to model.

### 3. Design of Recurrent Neural Network Controller with Feedback

In this paper, we propose a recurrent NN controller with the feedback structure. Method of Optimal selection to the appropriate control gain is follows that it makes a inference gain needed for the next stage using inference control input and inference error between the desired reference of the past and the present output of plant. We shown the structure of proposed NN controller in Fig. 1

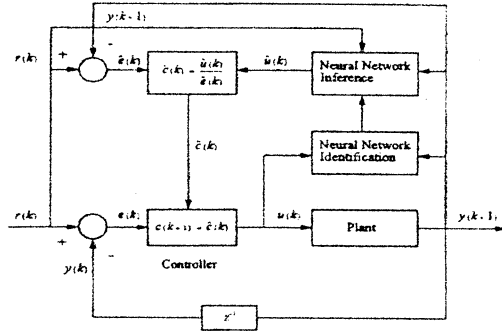


Fig. 1 Structure of feedback recurrent NN controller

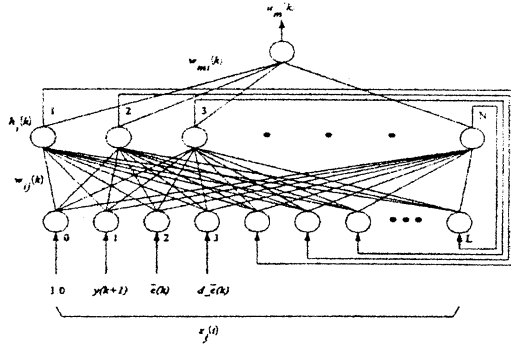


Fig. 2 Learning structure of recurrent NN

### 4. Feedback Recurrent Neural Network

Proposed NN can implement the dynamic characteristic of system through the feedback repeatedly delayed output and output of hidden layer to input stage. We represent learning structure of a recurrent neural network in Fig.2. Transfer characteristic of signal and initialization follows as.

Input  $z_j(k)$  of given NN consist of plant output  $y(k+1)$ , error  $\bar{e}(k+1)$ , change ratio of error  $d-\bar{e}(k)$  and feedback input from hidden layer  $h_{(j-n)}(k)$ , bias  $z_0$ .

Output of hidden layer  $h_i(k)$  for each learning input describe as

$$h_i(k) = S_i(I_i(k)), \quad i = 1, 2, \dots, N \quad (28)$$

In above equation ,

$$I_i(k) = \sum_{j=0}^L \omega_{ij}(k) z_j(k), \quad L = 3+N, \quad S_i(I_i(k)) = \exp(-s_i(I_i(k)))^2 \quad (29)$$

Where, L and N is number of each input and hidden unit,  $\omega_{ij}$  denote a weight vector connect input layer and hidden layer, hidden layer has a characteristic of Bell function and where  $s_i$  imply a gradient of Bell function and  $z_0$  denote bias to decide the position of the Bell function. Final output of neural network  $u_m(k)$  is

$$u_m(k) = \sum_{i=1}^N \omega_{mi}(k) h_i(k) \quad (30)$$

We were constructed that input units are 17 numbers, hidden units are 14 numbers and output unit is 1 number in Fig. 2 and weight vector  $\omega_{ij}$  and  $\omega_{mi}$  is initialized optionally by a number -0.5 ~ +0.5 and gradient of Bell function ( $s_i$ ) have 1.0, bias lied 1.0 at 0th unit of input layer. Proposed learning rule of recurrent NN is as follows

Error function of output layer  $J(k)$  is difference of plant input  $u(k)$  and actual output  $u_m(k)$  of NN

$$J(k) = \frac{1}{2} \sum [u(k) - u_m(k)]^2 \quad (31)$$

Gradient of error function  $J(k)$  for hidden layer and weight vector of output  $\omega_{mi}$  is shown.

$$\frac{\partial J(k)}{\partial \omega_{mi}} = \delta_m(k) \quad (32)$$

In this equation

$$\delta_m(k) = [u(k) - u_m(k)] h_i(k) \quad (33)$$

Gradient of error function for a weight vector  $\omega_{ij}$  connect input layer with hidden layer is

$$\frac{\partial J(k)}{\partial \omega_{ij}} = \sum_{n=1}^N \delta_n \gamma_{nij}(k) \quad (34)$$

In above equation ,  $\delta_n(k)$  and  $\gamma_{nij}$  is

$$\delta_n(k) = \sum_m \delta_m(k) \omega_{mi} \quad (35)$$

$$\gamma_{nij}(k) = \frac{\partial h_n(k)}{\partial \omega_{ij}} =$$

$$S'_n(I_n(k)) \left( \delta_n z_j(k-1) + \sum_{p=1}^N \omega_{n(p+n)} \gamma_{pny}(k-1) \right) \quad (36)$$

$\gamma_{nij}(0) = 0$ , in above equation ,  $\delta_m$  is Kronecker delta and N is numbers of recurrent unit that feedback is created in hidden layer .

$$\omega = [\omega_{ij}, \omega_{mi}]^T \quad (37)$$

Weight vector of proposed NN can be derived through the matrix equation. Hence, learning rule of NN is obtained

$$\omega^{new} = \omega^{old} - \eta \nabla J(k) \quad (38)$$

Where  $\eta$  denote a ratio of learning, learning rule in above, consist a formula of repetition added a feedback characteristic in a error back propagation algorithm of in existing. Generally, though it take

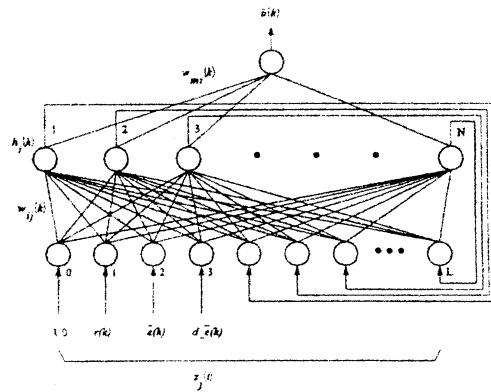


Fig 3 Structure of recurrent NN for inference of a control input

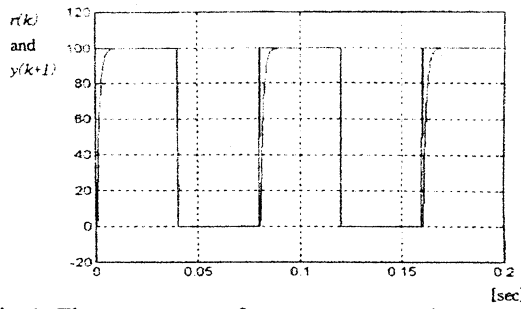


Fig 4 The response of recurrent neural network controller with feedback structure applied in plant equation  $y(k+1) = 1.39711y(k) - 0.538894 y(k-1) + 2.59383u(k)$

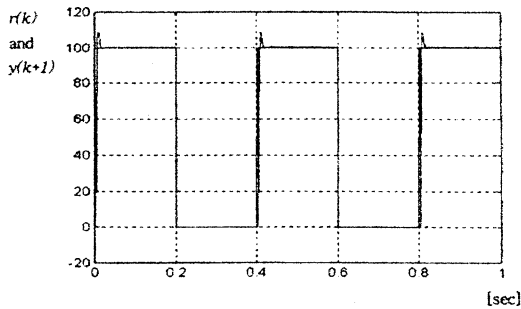


Fig 5 The response of recurrent neural network controller with feedback structure applied in plant equation  $y(k+1) = 0.8607 y(k) + 0.0792u(k) + 0.0601u(k-1)$

many time to learning according to large scale of NN but we can solve that multi NN of existing have a limit to implement a dynamic characteristic of system. We shown in Fig. 3 that NN inferring a expected control input from learned NN.

Input  $z_j(k)$  consist of a reference value  $r(k)$ , error  $\bar{e}(k)$ , change ratio of error  $d-\bar{e}(k)$  and feedback input from the hidden layer  $h_{j-n}(k)$  and bias  $z_0$ .

Output of hidden layer for input  $h_i(k)$  is derived

from equation (28), (29) and final output  $\bar{u}(k)$  of NN in order to infer control input is

$$\bar{u}(k) = \sum_{i=1}^N \omega_{mi}(k) h_i(k) \quad (39)$$

Where, all structure of NN for the weight vector of hidden layer and output layer, number of input unit, number of hidden unit is same as the recurrent neural network in Fig.2

## 5. Conclusions

Proposed NN use adding to input component make feedback a output of hidden layer in NN structure of general three layer.

This NN can implement a dynamic characteristic of system make repeatedly a feedback to input stage the delayed input and input of the hidden layer. NN controller of feedback structure configure with identification of recurrent NN and learning of recurrent NN in order to infer demanded information of a control input. We were confirmed by simulation that designed recurrent NN controller have a stable response without a oscillation in transient area and a learning time of Fig.4 show more faster than learning time of Fig.5.

## References

- [1] R.J.Williams and D.Zipser : A Learning Algorithm for Continually Running Fully Recurrent Neural Networks Vol.1,No.2, pp 270-280, 1989.
- [2] B.A.Pearlmutter : Learning State Space Trajectory in Recurrent Neural Networks, Neural Computation, Vol.1, No2, pp263-269, 1990.
- [3] Y.Fang and T.J.Sejnowski : Faster Learning for Dynamic Recurrent Back Propagation, Neural Computation, Vol.2,No3, pp270-273, 1990.
- [4] R.J.Williams : Training Recurrent Networks Using the Extended Kalman Filter, Baltimore, Maryland,Proc. of IEEE Int. Joint Conf. On Neural Networks, Vol 4, pp241-246,1992.
- [5] D.Chen, C.L.Giles, G.Z.Sun, H.H.Chen, Y.C.Lee and M.W.Goudreau : Constructive Learning of Recurrent Neural Networks, Sanfransisco, California, Proc.of Int. joint Conf. On Neural Networks pp1196-1201,1993.
- [6] S.S.Rao and V. Ramamurti : A Hybrid Technique to Enhance the Performance of Recurrent Neural Networks for Time Series Prediction, Sanfransisco, California,Proc.of IEEE Int. Joint Conf. on Neural Networks, pp.52-57,1993.

# ELECTRICAL EQUIVALENT CIRCUIT AND RESTING MEMBRANE POTENTIAL OF NEURON

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** New equivalent circuits of neuron, different from the equivalent circuit of Hodgkin-Huxley, are proposed. In these circuits, ion-condensers are a concept to represent the equilibrium potential of ions, and power sources, to represent ion-pumps. Using these equivalent circuits, we derive equations which relate resting membrane potential with capacity of ion-pumps, intracellular impermeant anions concentrations, extracellular ions concentrations and permeability of ions. Using these equations, we evaluate the influence of presence of intracellular impermeant anions on the resting membrane potential.

**Keywords** cell, resting membrane potential, equivalent circuit, ion pump, ion-condenser, impermeant anion

## 1. INTRODUCTION

An accurate equivalent circuit of neuron would not only help to analyze electrical characteristics of the neuron, but also constitute another basis of study on neural network systems, artificial intelligence, etc. There is much research on mechanism of excitation and signal transmission of neuron accounted by the equivalent circuit of Hodgkin-Huxley. However, in Hodgkin-Huxley equivalent circuit, important factors such as intracellular impermeant anions and ion-pumps have not been considered<sup>(1)-(3)</sup>. Thus, to understand neuron more accurately, we propose new equivalent circuits consisted of ion-condensers, a concept to represent the equilibrium potential of ions, and power sources to represent ion-pumps<sup>(4)</sup>.

Any cell has a resting potential across its membrane due to presence of ion-pumps and intracellular impermeant anions. However, a relationship between ion-pumps and impermeant anions has not been taken into account yet. In this paper, we propose equations to calculate membrane potential by relating intracellular impermeant anions concentrations, extracellular ions concentrations, membrane ions permeability, and active ions currents generated by ion-pumps in the equilibrium state.

## 2. ION-CONDENSER

Hodgkin-Huxley's equivalent circuit uses power sources to represent equilibrium potential of ions. However, as a cell continues being excited, ions concentrations gradients become lower, namely, the equilibrium potential of ions decreases. What recovers the ions concentrations gradients are the ion-pumps<sup>(5)(6)</sup>. In order to find how the resting potential relates with ion-pumps and intracellular impermeant anions,

permeant ions concentrations gradients, namely, the equilibrium potential of ions, have to be defined as variables. In this case, using constant voltage power sources to represent the equilibrium potential of ions as in Hodgkin-Huxley's equivalent circuit is not suitable. Assuming the extracellular ions concentrations as constant, the concentration gradient of an ion across cell membrane, namely the equilibrium potential, depends on the quantity of the ion in the cell and volume of the internal fluid. This characteristic appears in a condenser, in which voltage depends on quantity of electrical charge and area of the two poles. Thus, in this paper, we introduce a type of condenser to represent ions concentration gradients, and its voltage to represent the equilibrium potential. We name this condenser as an *ion-condenser*.

The equilibrium potential  $V_X$  of an ion X is obtained from Nernst relation

$$V_X = -\frac{RT}{z_X F} \ln \frac{[X]_i}{[X]_o} \quad \dots\dots\dots (1)$$

where  $[X]_i$  and  $[X]_o$  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.

Since the potential of an ion-condenser is determined by the ion concentration gradient, the electrical charge of an ion-condenser can be thought as generated by the ion concentration gradient. Quantity of the electrical charge is calculated according to the following equation

$$q_X = z_X v ([X]_i - [X]_o) \quad \dots\dots\dots (2)$$

where  $v$  is volume of the internal fluid of the cell.

From eqs.(1) and (2), we get the relationship between voltage and electrical charge of an ion-condenser



$$V_X = -\frac{RT}{z_X F} \ln \left( \frac{q_X}{v z_X [X]_o} + 1 \right) \dots\dots\dots (3)$$

Equation (3) shows that the characteristics of an ion-condenser is different from the normal condenser ( $V = q / C$ ), i.e., an ion-condenser is a nonlinear condenser. The characteristics of an ion-condenser depends on extracellular concentration of the ion, volume of the internal fluid of the cell, temperature, and the ion valence.

To ease calculation and discussion, we define the total electrical charge of an intracellular ion X as

$$Q_X = z_X v [X]_i \dots\dots\dots (4)$$

From eqs. (1) and (4), we get the relationship between  $Q_X$  and  $V_X$

$$Q_X = z_X v [X]_o e^{-\frac{z_X F}{RT} V_X} \dots\dots\dots (5)$$

### 3. EQUIVALENT CIRCUIT OF A CELL INCLUDING ION-PUMPS

Figure 1 shows how we synthesized an electrical equivalent circuit of a cell membrane. Figure 1(I) shows fluids in different concentrations of two types of ions separated by a membrane. In Fig.1(I) electrochemical potential causes the ions to diffuse through the membrane. To structure the equivalent circuit of the membrane, we represented the electrical capacity of the membrane as a condenser and the permeability of the membrane as conductance, so circuits were drawn to equal the permeability through membrane as shows Fig.1(II). If we represent the chemical potential of the ions as the electrical potential of an ion-condenser, Fig.1(II) can be drawn as Fig.1(IV).

Figure 2 shows the flux of ions of a cuttlefish giant neuron<sup>(4)-(6)</sup>. Here we assume that the ion-pumps in the cell are only  $\text{Na}^+/\text{K}^+$ -pumps, and the extracellular permeable ions are  $\text{Na}^+$ ,  $\text{K}^+$ ,  $\text{Ca}^{++}$ ,  $\text{Mg}^{++}$ , and  $\text{Cl}^-$ .  $\text{Y}^-$  represents the sumation of possible impermeant anions. We use conductances to represent ion-channels and permeability of membrane as in Hodgkin-Huxley's equivalent circuit<sup>(1)-(3)</sup>. Since the ions currents of  $\text{Na}^+/\text{K}^+$ -pumps are related to the concentration of intracellular  $\text{Na}^+$ , we can consider the  $\text{Na}^+/\text{K}^+$ -pumps as 2-current-power sources, in which the output currents are related to the intracellular concentration of  $\text{Na}^+$ . From the viewpoint of electrical engineering, current-power sources of ion-pumps are connected in parallel with ion-channels, and in series with ion-condensers. From Fig.1(IV) we draw the equivalent circuit of Fig.2 as Fig.3, where  $C_{\text{Na}}$ ,  $C_{\text{K}}$ ,  $C_{\text{Ca}}$ ,  $C_{\text{Mg}}$ , and  $C_{\text{Cl}}$  represent ion-condensers of  $\text{Na}^+$ ,  $\text{K}^+$ ,  $\text{Ca}^{++}$ ,  $\text{Mg}^{++}$ , and  $\text{Cl}^-$ ; equivalent conductance  $g_{\text{Na}}$ ,  $g_{\text{K}}$ ,  $g_{\text{Ca}}$ ,  $g_{\text{Mg}}$ , and  $g_{\text{Cl}}$  represent permeability through membrane of  $\text{Na}^+$ ,  $\text{K}^+$ ,  $\text{Ca}^{++}$ ,  $\text{Mg}^{++}$ , and  $\text{Cl}^-$ ; and  $C_m$  represents the membrane electrical capacity.  $I_{\text{PNa}}$  and

$I_{\text{PK}}$  are currents of  $\text{Na}^+$  and  $\text{K}^+$  actively transferred by  $\text{Na}^+/\text{K}^+$ -pumps;

From Fig.3 we obtain a general equivalent circuit to represent possible types of cells as shows Fig.4, where  $X_i$  ( $i = 1, 2, \dots, n$ ) represent permeable ions in the external fluid, and  $I_{\text{PX}i}$  represent active transport currents of ion  $X_i$ . If  $X_i$  has not active transport,  $I_{\text{PX}i} = 0$ .

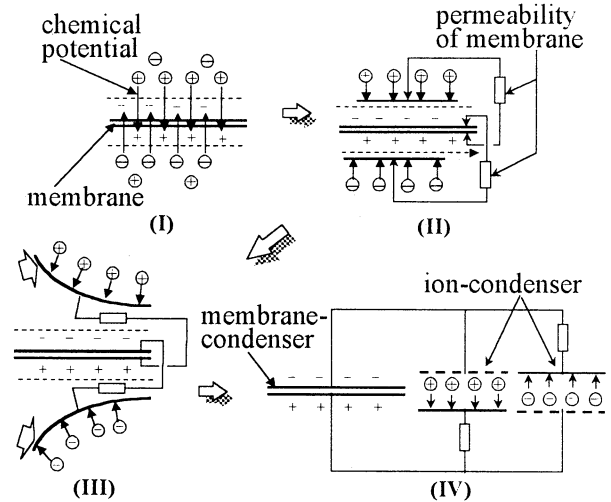


Fig.1 Relationship between ion-condenser and membrane condenser

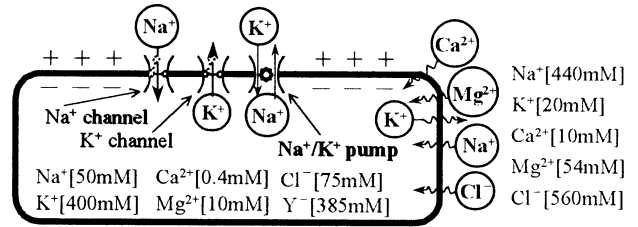


Fig.2 Isolated giant neuron in the external fluid<sup>(5)(6)</sup>

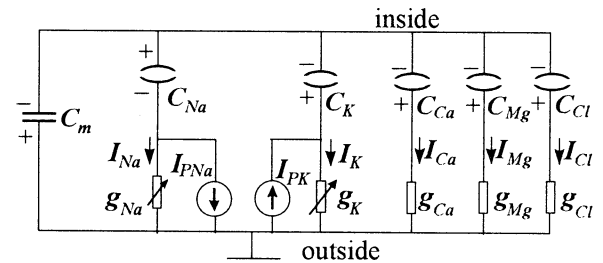


Fig.3 Equivalent circuit of isolated giant neuron including ion-pumps

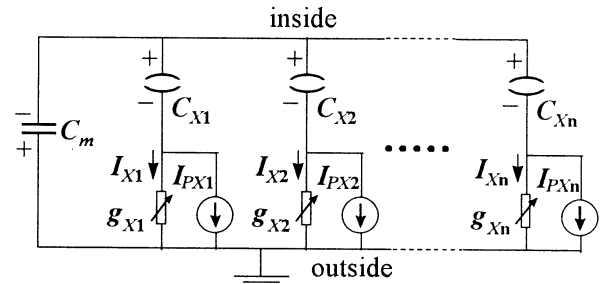


Fig.4 A general equivalent circuit of a cell

#### 4. INFLUENCE OF IMPERMEANT ANIONS ON RESTING MEMBRANE POTENTIAL

It is known that impermeant anions affect the resting membrane potential. However, it has not been developed any clear equation to calculate a relationship between resting membrane potential and impermeant anions concentrations. In this section, we discuss how impermeant anions influence our equivalent circuit.

Let us consider the flow of ions through a cell membrane when the cell has intracellular impermeant anions. Since the extracellular ions concentrations can be assumed as constant, we discuss the process of an imaginary cell, which has a permeable cation  $K^+$  and an impermeant anion  $A^-$  as intracellular ions, put into a  $K^+$ -and- $Cl^-$  solution. Figure 5 shows that the presence of impermeant anions leads to a membrane potential.

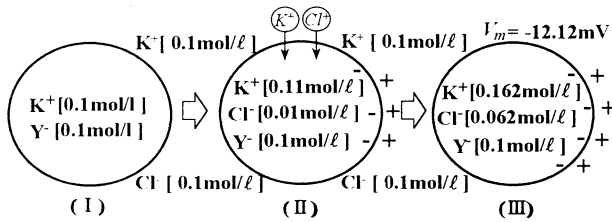


Fig.5 Ion exchange through cell membrane

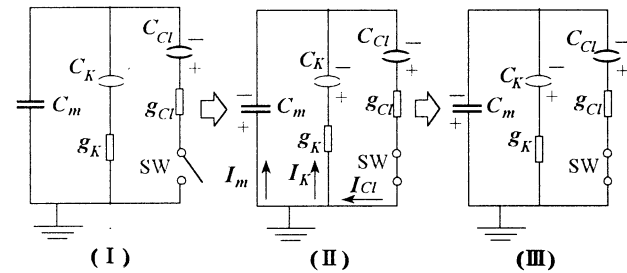


Fig.6 Ion exchange through cell membrane represented by an equivalent circuit

The instant the cell is put into solution as shows Fig.5(I), an equivalent circuit can be drawn as shown in Fig.6(I), where  $C_{Cl}$  and  $C_K$  represent ion-condensers of  $Cl^-$  and  $K^+$ ,  $g_{Cl}$  and  $g_K$  represent equivalent conductance of permeability through membrane of  $Cl^-$  and  $K^+$ , and  $C_m$  represents membrane electrical capacity. In this state, external  $Cl^-$  has chemical potential to flow into the cell. To represent the chemical potential in the equivalent circuit,  $Cl^-$ -ion-condenser has the same equivalent potential. Similar to  $Cl^-$ , for  $K^+$  has not concentration gradient, the potential of  $K^+$ -ion-condenser is 0 mV. Figure.5(II) shows the state of the cell just after the cell is put into solution, where  $Cl^-$  flows into the cell by chemical potential and  $K^+$  flows into the cell by electrical potential. Figure.6(II) expresses the equivalent circuit, in which condenser  $C_m$  and ion-condenser  $C_K$  are being charged by ion-condenser  $C_{Cl}$ . Figure.5(III) shows the steady state of the cell, and Fig.6(III) shows the

equivalent circuit of Fig.5(III). Figure 6(III) shows that  $C_m$ ,  $C_K$  and  $C_{Cl}$  have the same potentials in the steady state. Figure 6 shows that impermeant anions can cause membrane potential, however, it does not change the structure of the equivalent circuit.

#### 5. EQUATION OF RESTING MEMBRANE POTENTIAL

For impermeant anions do not influence the structure of the equivalent circuit, in this section we calculate the resting membrane potential using the equivalent circuit of Fig.4.

Since the external fluid can be thought as neutral, volume of electrical charge  $q_m$  concentrated on the membrane is equal to the total electrical charge of intracellular ions. If we represent the electrical charge of a permeable ion  $X$  as  $Q_X$ , and of an impermeant anion  $Y$  as  $Q_Y$ , the following equations are written

$$\sum Q_X + \sum Q_Y = q_m \quad \dots\dots\dots (6)$$

$$q_m = C_m V_m \quad \dots\dots\dots (7)$$

$$Q_Y = v z_Y [Y]_i \quad \dots\dots\dots (8)$$

where  $V_m$  is the membrane potential,  $C_m$  is the membrane electrical capacity,  $v$  is volume of the internal fluid of the cell and  $z_Y$  is valence of  $Y$ . From the equivalent circuit of Fig.4, we obtain the following equations:

$$V_m = \frac{I_X}{g_X} + V_X \quad \dots\dots\dots (9)$$

$$I_X + I_{PX} = \frac{dQ_X}{dt} \quad \dots\dots\dots (10)$$

$$I_m = \frac{dq_m}{dt} \quad \dots\dots\dots (11)$$

In the steady state, the concentrations of ions are constant. Thus, from eq.(10) we obtain

$$I_X = -I_{PX} \quad \dots\dots\dots (12)$$

Hence, eq.(9) becomes

$$V_X = V_m + \frac{I_{PX}}{g_X} \quad \dots\dots\dots (13)$$

And eq.(5) becomes

$$Q_X = z_X v [X]_o e^{-\frac{z_X F}{RT} \left( V_m + \frac{I_{PX}}{g_X} \right)} \quad \dots\dots\dots (14)$$

Because the electrical charge on membrane  $q_m$  is far smaller than  $Q_X$ , using eqs.(14) and (8) we obtain the following equation from eq. (6)

$$\sum \left( z_X [X]_o e^{-\frac{z_X F}{RT} \left( V_m + \frac{I_{PX}}{g_X} \right)} \right) + \sum z_Y [Y]_i = 0 \quad \dots\dots\dots (15)$$

Using eq.(15) we can calculate the resting membrane potential  $V_m$ . Equation (15) shows that the resting membrane potential is determined by 1) concentrations of extracellular ions and their valences, 2) intracellular impermeant anions concentrations, 3)

active transport currents, i.e., capacity of ion-pumps, 4) permeability of ions which have active transport through membrane, and 5) temperature around the cell.

## 6. RESTING MEMBRANE POTENTIAL OF CUTTLEFISH GIANT NEURON WITHOUT OPERATION OF ION-PUMPS

Here we discuss the influence of impermeant ions on the resting membrane potential in case ion-pumps are stopped for a cuttlefish giant neuron.

Table 1 shows approximate ions concentrations in the axoplasm and in the external fluid of freshly isolated giant neurons<sup>(1)</sup>. From eq.(15) we obtain the following equation:

Ion	Concentration in axoplasm (mM)	Blood (mM)	Sea water (mM)
K <sup>+</sup>	400	20	10
Na <sup>+</sup>	50	440	460
Ca <sup>2+</sup>	0.4	10	10
Mg <sup>2+</sup>	10	54	53
Cl <sup>-</sup>	40~150	560	540
Impermeant anion (Y <sup>-</sup> )	385	—	—

**Table 1** Approximate concentrations of ions in the axoplasm of freshly isolated giant neurons and in the external fluid<sup>(1)</sup>

$$2\left([Ca^{++}]_o + [Mg^{++}]_o\right)e^{-\frac{2F}{RT}V_m} + \left([Na^+]_o + [K^+]_o\right)e^{-\frac{F}{RT}V_m} - [Cl^-]_o e^{\frac{F}{RT}V_m} + \sum(z_Y[Y^-]_i) = 0 \quad \dots\dots\dots(16)$$

Hence

$$[Cl^-]_o e^{\frac{3F}{RT}V_m} - \sum(z_Y[Y^-]_i)e^{\frac{2F}{RT}V_m} - \left([Na^+]_o + [K^+]_o\right)e^{\frac{F}{RT}V_m} - 2\left([Ca^{++}]_o + [Mg^{++}]_o\right) = 0 \quad \dots\dots\dots(17)$$

This is a cubic equation with respect to  $e^{\frac{F}{RT}V_m}$ . From Table 1,

$$\sum[Cl^-]_o = 560mM, \quad \dots\dots\dots(18)$$

$$\sum(z_Y[Y^-]_i) = 385mM, \quad \dots\dots\dots(19)$$

$$[Na^+]_o + [K^+]_o = 460mM \quad \dots\dots\dots(20)$$

$$[Ca^{++}]_o + [Mg^{++}]_o = 64mM \quad \dots\dots\dots(21)$$

Hence eq.(17) becomes

$$-\left(e^{\frac{F}{RT}V_m}\right)^3 - 0.6875\left(e^{\frac{F}{RT}V_m}\right)^2 + 0.8214\left(e^{\frac{F}{RT}V_m}\right) + 0.2286 = 0 \quad \dots\dots\dots(22)$$

From eq.(22) we obtain the value of the membrane potential in the steady state. At temperature 18 °C,  $V_m$  is -6.8 mV. Since the normal resting

membrane potential of a cuttlefish giant neuron is -61mV, the influence of intracellular impermeant anions on the membrane potential is very small. This result validates the experiment in which, after stopping the ion-pumps, the membrane potential decreases slowly until a small value<sup>(6)</sup>.

## 7. CONCLUSION

This paper proposed a new equivalent circuit of neuron by introducing the concepts of ion-condensers and current-power 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 the resting membrane potential. The results are the following:

- I. The resting membrane potential of a cell is determined by
  - (a) concentrations of extracellular ions grouped by valences;
  - (b) intracellular impermeant anions concentrations;
  - (c) capacity of ion-pumps;
  - (d) permeability of ions transported by ion-pumps through membrane;
  - (e) and temperature around the cell.
- II. The influence of intracellular impermeant anions on the resting membrane potential is very small in actual neurons.

This study was supported by Grant-in-Aid for Encouragement of Young Scientists of the Ministry of Education .

## REFERENCES

- (1) Hodgkin AL(1962), Ionic movements and electrical activity in giant nerve fibres. The Croonian Lecture, Proc. Roy. Soc. B, vol. 148, pp.1-37
- (2) Hodgkin AL, Huxley AF(1952), A Quantitative Description of Membrane Current and Its Application to Conduction and Excitation in Nerve. J. Physiol. 117, pp.500-544
- (3) Hodgkin AL, Huxley AF(1952), Currents Carried by Sodium and Potassium Ions Through The Membrane of The Giant Axon of Loligo. J. Physiol. 116, 449-472
- (4) Zhang X, Wakamatsu H(1998), Electrical Equivalent Circuit and Resting membrane Potential of Neuron(in Japanese). Proc. Electron. Syst. Conf., IEE of Japan, pp.183-188
- (5) Hall ZW(1992), Molecular Neurobiology, Sinauer Associates. Inc., New York
- (6) B. Alberts et al. (1994), Molecular biology of the cell, 3rd Edition, Garland Publishing, Inc., New York

# An Adaptive Associative Memory System based on Autonomous Reaction between Image Memories

Yasuo Kinouchi\* , Masahiro Mizutani\* , Shoji Inabayashi\*\* , Akira Satou\* , and  
Fumitoshi Shouji\*

**Abstract**— To construct a 'thinking-like' processing system, an architecture of an adaptive associative memory system is proposed. This memory system treats 'images' as basic units of information, and adapts to the environment of the external world by means of autonomous reactions between the images. The images do not have to be clear, distinct symbols or patterns, they can be but also ambiguous, indistinct symbols or patterns as well. This memory system is a kind of neural-network made up of nodes and links, called a localist spreading activation network. Each nodes holds one image based on localist manner. Images in high-activity nodes interact autonomously and generate new images and links. By this reaction between images, various forms of images are generated automatically under constraints of links with adjacent nodes in the associative memory system. Simple operations of the image reaction are proposed. Each operation generates a new image by combining pseudo figures or features and links of two images.

**Keywords**— adaptive system, associative memory, neural logic, image, autonomous reaction

## I. INTRODUCTION

The action of 'thinking', which we do daily, can be recognized as a kind of information processing. The purpose of this paper is to describe a fundamental method for carrying out the information processing action of 'thinking' on a computer. However, our intention is not to describe how the brain operates, but how the fundamental structure and architecture of an information processing system can function like a brain. To construct such a system, we are developing an adaptive associative memory system. This memory system treats 'images' as basic units of information, and adapts to the environment of the external world by means of autonomous reactions between the images. The images do not have to be clear, distinct symbols or patterns, they can be but also ambiguous, indistinct symbols or patterns as well. To construct a 'thinking-like' processing system, ambiguity and polysemy of the basic information units play an important role.

The associative memory system we are developing is a kind of neural-network made up of nodes and links, called a localist spreading activation network. Each nodes holds one image based on localist

manner. When an image is input to the associative memory system, a group of images called extracting images, which are used for extracting features from the input image, transmit stimulation signals of the features of the input images to other images. As a result of spreading activation, the network outputs images from the highest activity nodes. This system learns by reinforcement learning based on how effective the output image is[1].

In addition, reactions between nodes are introduced. Images in high-activity nodes interact autonomously and generate new images and links. By this reaction between images, various forms of images are generated automatically under constraints of links with adjacent nodes in the associative memory system. Images are also subject to natural selection, for instance, images in high-activity nodes survive while images in low-activity nodes gradually die out. Based on reactions between images and the rule of natural selection, this associative memory system can 'remember' appropriate images, and gradually adapt to the environment.

A simple reaction model consisting of three operations (addition, subtraction, and multiplication) is proposed. If the information unit is a visual image, it can be decomposed into several elementary pseudo figures or features. Each operation generates a new image by combining pseudo figures or features and links of two images.

We call the adaptive associative memory system proposed here 'Image Reactor' from the fact that the images in the system react to each other. This paper discusses the design concept, functions and configuration of the image reactor. To evaluate the image reactor, we are developing a simulation model of a robot that uses the image reactor as its 'brain' and walks around a miniature garden.

## II. DESIGN CONCEPT OF IMAGE REACTOR

The action of "thinking" includes the following three features

- Images are used as basic processing units when we think.
- Thinking related to the complicated associative memory function of the image.
- New images are created from memorized images when thinking.

\* Tokyo University of Information Sciences, 1200-2 Yatocho Wakaba-ku Chiba-shi, 265-8501 JAPAN, kinouchi@rsch.tuis.ac.jp, mizutani@rsch.tuis.ac.jp

\*\* System Sogo Kaihatsu Co., Ltd. 8-4-19 Tajima Urawashi, 336-0037 JAPAN, ina@sskco.co.jp

These features, as well as the following two points, were taken into consideration in the design of the image reactor.

- We adapt to externalities through ‘thinking’.
- By making up a model of a externality inside the processing mechanism, we adopt externality.

The information processing system we are developing incorporates these features, and is examined below from the viewpoint of how information is represented, organized, and generated or changed.

### A. Representation of Information

#### A.1 Local Representation

The method for representing information can be either local representation or distributed representation. We adopted local representation for the image reactor. In distributed representation, data handling and setting up a large-scale storage system are difficult. In local representation, both are easy[2]. Moreover, language, symbols, and images can be easily expressed by the same method. Image data in the image reactor are stored in nodes so that local representation is possible.

#### A.2 Features of the Image as Information Representation

Information in the nodes is not only indistinct image, it also has the following properties.

1. Separability: Images that are separable can be divided into smaller images that show a part or feature of the larger image, but that can still be identified as a feature or a part of the larger image. For example, “p” and “pen” are symbols without any obvious relationship. In this case, “pen” is not separable. Considering a picture of a pen, however, such an image can be separated into images of partial features of the original pen, such as images of the nib and grip. This case is an example of information representation with separability.
2. Connectivity: Connectivity is the opposite of separability. New images can be created by spatially combining several images as a unit. Multidimensional combination is also possible.
3. Stability (Redundancy): The meaning of what is being represented does not change much if its form changes within a certain range. That is to say, the deformation does not change the meaning of the image in the representation very much. In the example of the image of the pen, the meaning of the image changes little when the length and thickness of the grip are changed to some extent.

In this paper, information that satisfies the above properties is treated as an image, although at first

glance it may not seem like an image. For example, if a subset of a set of features of something satisfies three properties described above, then the subset can be treated as an image. The image is connected with symbols and words through these three properties, because the features are a kind of attribute.

### B. Organization of Information

Distributed representation has been widely studied as the neural logic for an associative memory system[3]. However, we use local representation in this system. In our local representation the image is stored in the node. The links which connect images are weighted and the association between images is controlled by the change in the weight of the links. Local representation is adopted for two reasons. First, since the content of the memory and the method for making associations are directly connected in the distributed representation method, it is difficult to change the memory content and the method of association independently. The main advantage of the Neumann-style computer is that addresses can control the memory regardless of the content of the memory. By enabling the content and the association (access) method of the memory to be changed independently, large-capacity associative memory system can be easily set up. Second, the combination of multiple associative memory system using local representation is easier than the associative memory system using distributed representation.

### C. Generation and Change of Information

New images are created by the autonomous reactions between images and old images disappear by natural selection of images. The image reactor adapts to the environment by creating effective images for carrying out actions and by eliminating the ineffective images. The autonomous reaction of the image in the associative memory continually generates new images in the image reactor. According to the action potential of an image, each image randomly interacts with other images to generate new images. New links are formed between new images and old images. This process is called a reaction between images. Local representation makes it easy for images to react directly with each other in the associative memory. Images which are useful for adapting to the environment are kept while unnecessary images disappear, and the new and old images operate as a part of the associative memory system. The remaining, useful images constitute a part of the image model of the environment in the associative memory system.

The genetic algorithm is widely known as a

method to form the new elements, which it does by random interaction of individual elements[4]. However, it does not consider the relationship between elements. In the image reactor, the relationship between elements(images) is realized through the link in the image reactor.

### III. IMAGE MEMORY NETWORK OF THE IMAGE REACTOR

The robot simulation, which uses an image reactor for the robot's "brain", was set up along the lines of computational neuroethology[5]. Through an analysis of the robot's actions, we can learn how the image reactor functions.

#### A. Configuration of the Image Reactor and its Environment

A world composed of a robot and a miniature garden was set up. The miniature garden contains various obstacles and food. The robot moves around the garden until it finds and eats the food as images of scenes of the garden are input. Features of the image of the present scene remind the robot of an image of a scene which the robot has experienced in the past. The robot adopts an action which is related to the remembered image. However, the image of the scene which is remembered is not simply an image of a scene of the miniature garden which the robot experienced in the past. Through reaction and natural selection of images, these images are changed to key images. The key images are of effective marks in the garden which indicate the likelihood or possibility of food being located nearby. The image reactor forms marks in the adaptive action. In the action of the food search, the robot carries out concept generation by generating these key images. The robot's function modules are shown in Fig.1.

1. Input module: Input module converts information of a scene of the miniature garden into image data which the robot can understand as information.
2. Motor module: Motor module moves the robot inside the miniature garden, according to the decision of the image reactor.
3. Image reactor: Image reactor corresponds to the 'brain' of the robot. It is composed of an image memory network and a control module, as shown in Fig.1. The control module controls the image memory network which memorizes the image, and association, and reaction are carried out.

#### B. Image Memory Network of the Image Reactor

The basic elements of the image memory network are nodes and links. Each node has an image data.

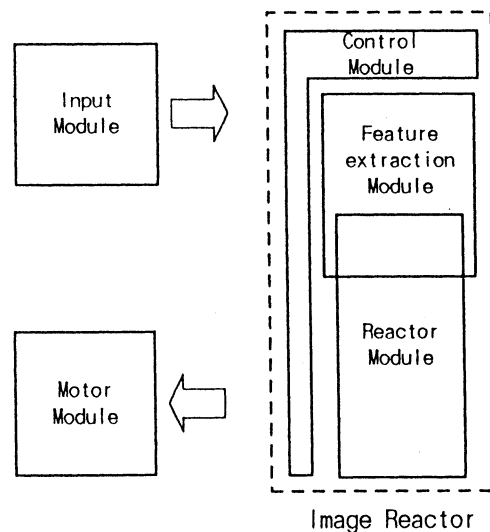


Fig. 1. Function Module of Robot

The condition of the node is defined by the level of action potential and degree of excitement. The node transmits stimulation signals in one direction to other node. The image memory network is composed of a feature extraction module and reaction module to fulfil the associative memory function. The feature extraction module extract features from the input image data and stimulates nodes in the reactor module through the links as shown Fig.2. It has a large number of nodes for extracting features which is done by comparing the input image data and its own image data, and individually evaluating the resemblance. If the resemblance is high, these feature extraction nodes send a stimulation signal through the link. Images in the reaction module receive the signals from multiple feature extraction nodes and calculate the total value of the stimulation signals which takes into consideration the weight of the link each signal propagates along. Depending on the value of the stimulation signals, the node becomes excited. Nodes which have image data that are similar to input image data easily become excited.

There are two types of images in the feature extraction module. One type is included only in the feature extraction module, other type is included in both the feature extraction module and the reactor module. When the action potential of an image rises, the image in the reaction module moves to the feature extraction module and extracts features from the input image. When the action potential is low, it is excluded from the feature extraction module. Feature extraction is carried out in both images prepared before the start of the simulation, as well as images for which the action potential is high.

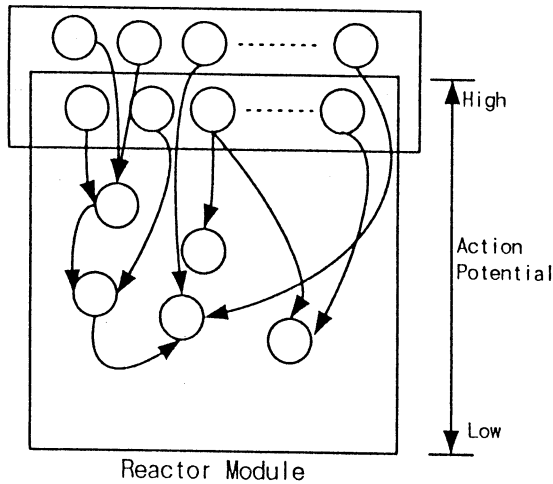


Fig. 2. Feature Extraction

#### IV. REACTION AND SELECTION OF IMAGES

The images in the reaction module generate new nodes and link by autonomous reaction between nodes. Images with high action potential mutually and randomly react to each other. The input of a reaction is a set of two images and links attached to them. The output is a new image and new links to the new image from old images. Whether or not an image reacts is determined by the relative level of the action potential of the image. Generally, the number of images reacting simultaneously is large. The reaction between images can involve a wide variety of complex operations, however for this evaluation, we assume three basic operations.

##### A. Reactions between Image Data

Features or pseudo features are usually spatial configurations of the image data. The image data are expressed as an arrangement of features or pseudo features which may be simple figure or a part of a feature. The property of separability, connectivity, stability discussed above are satisfied when the image is made up of a configuration of features and pseudo features. Reactions between image data take the form of the operations of addition, or subtraction, multiplication. These reaction operations between input image data  $I_a$  and  $I_b$  of node  $A$  and  $B$  are defined as follows, generating a output node  $C$  with image data  $I_c$ :

*Addition* :  $I_c = I_a + I_b$ .

$I_c$  has features  $I_a, I_b$  retain.

*Subtraction* :  $I_c = I_a - I_b$ .

$I_c$  has features of  $I_a$ , but not of  $I_b$ .

*Multiplication* :  $I_c = I_a * I_b$ .

$I_c$  has features common to both  $I_a$  and  $I_b$ .

The link to the node  $C$  from others is created as follows.

*case of Addition* : Links connected to  $A$  or  $B$  are also links to  $C$ .

*case of Subtraction* : Links connected to only  $A$  are link to  $C$ .

*case of Multiplication* : Links connected to only both  $A$  and  $B$  are link to  $C$ .

##### B. Selection and Reinforcement Learning

The image reactor changes the action potential and weights of links based on the Hebb rule, and learns based on reinforcement manner. The action potential of the node increases when it is excited and decreases when it is not excited. The node(image) disappear, when the action potential of the node drops below certain threshold. The weight of the link approaches zero when stimulation signals do not propagate in the link. Nodes disappear at a rate almost equal to the rate new images are generated.

The image reactor carries out reinforcement learning when the food is found. This is done by giving the nodes a higher potential in proportion to the actual level of the potential. In effect, the potential can be used as a scale which shows how much time has passed since the nodes was excited.

#### V. CONCLUSION

The architecture of the associative memory system with the adaptation function was described. Features of this architecture are that it facilitates the organizing of information by adopting local representation and it adapts to externalities while images are created by reactions of the images. In the future, we plan to simulate a whole image reactor and clarify the operating characteristics.

#### ACKNOWLEDGMENTS

We thank Shuji Hashimoto(Waseda University) for helpfull discussions.

#### REFERENCES

- [1] Kosslyn, S.M., *Image and Brain*, MIT Press(1994).
- [2] McClelland, J.L. and Rumelhart, D.E., *Parallel Distributed Processing vol.1*, p77-109, MIT Press(1986).
- [3] Hassoun, M.H., *Associative Neural Memories -Theory and Implementation*, Oxford University Press(1993).
- [4] Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley(1989).
- [5] Meyer, J.A., and Guillot, A., *Simulation of adaptive behavior in animals: Review and Prospect From Animal to Animals*, Proceeding of the First International Conference on Simulation of Adaptive Behavior(1991).

# Human-Face Recognition Using Neural Network with Mosaic Pattern

Hiroshi KONDO and Sasha Bin Abdul Rahman

(Kyushu Institute of Technology, Kitakyushu 804-8550 Japan)

## Abstract

A robust human-face recognition using neural networks with mosaic pattern is presented. In a human-face recognition a difference of the face expressions makes it difficult. An appropriate mosaic face, however, keeps the fundamental feature of the face without depending the difference of the expressions. First it is shown that the presented neural network works so well for such differences of the face expressions, i.e., a smiling, crying, and irritating faces. Furthermore up facing, right rotate, and left rotate faces are examined within  $\pm 15$  degrees in this work. And the perfect recognition rate is attained for twenty persons. The utilized network is a well known back propagation neural one with three layers.

keywords: Neural Network,  
Human-Face, Recognition

## 1. Introduction

Recently an advanced security system requires a fast human-face

recognition system. And in such a field a neural network is often used. The neural network, however, works well only for a predetermined front face. In the case where the object face is a smiling or a crying or irritating one the neural system has not worked well<sup>1, 2</sup>. Furthermore, it has not worked well also for an up-facing, or a down-facing, or a right or left-rotated face. Here a mosaic face pattern is utilized in order to solve such problems. Appropriate size mosaic pattern of a human face contains a basic feature of the face. And it is inherently robust. For example, it is well known the Lincoln's mosaic face<sup>3</sup>. We utilize such mosaic patterns as an input for the neural network with three layers. There is an excellent paper<sup>4</sup> for face recognition using a mosaic pattern. The paper, however, treats only the front face with 12×12 mosaic pattern. In the real situation there are many different face expressions, and hence it has been required that the neural network for such human face recognition is robust. We examine such a robust property of the neural network in some detail by using not



only several different face expressions but the rotated one.

## 2. Neural network

First the gray level of the input face image must be changed into the full scale dynamic range in order to reduce the variation of the input face data due to something like a different lighting. The region of the face for the neural network input data does not include the hair because of its daily variation. The number of the input layer of the neural network is 144 (12×12) and that of the output layer is 21 (20 different human-faces+1 (other face)). It is very important to make an other faces output neuron for such a recognition system because unknown person may come appear very often. We utilize an average mosaic pattern of 50 persons except the 20 object persons as a teacher pattern of the other faces. The employed neural network is a back propagation one with three layers. So, the number of the hidden layer must be chosen optimally. In this work it is 13 by using integer dynamic programming. With the number of input layer IN, the hidden layer unit  $H_j$  is expressed as

$$H_j = f_j \left[ \sum_{i=1}^{IN} W_{ih_{ji}} I_i \right] \quad (1)$$

Where  $I_i$  is the  $i$ th input value of the input layer, and  $W_{ih_{ji}}$  is the coefficient between the input layer

and the hidden layer.  $f_j(\cdot)$  is a logistic function;

$$f_j(x) = \frac{1}{1 + e^{-x}}. \quad (2)$$

Similarly with  $W_{ho_{ji}}$  the coefficient between a hidden layer and output one, the output  $O_j$  is

$$O_j = f_j \left[ \sum_{i=1}^{HN} W_{ho_{ji}} H_i \right]. \quad (3)$$

It is very important to introduce one more output neuron for unfamiliar face. We can also recognize such unfamiliar face by using only threshold value for output layer value without using any other neuron. But the former is preferable to the latter one in our many simulation. The error rate of the latter system is 15%. On the otherhand the former system has 0% error rate in our examples.

## 3. Simulation examples

Figures 1 and 2 show the example normal photo and its teacher's mosaic pattern. Figures 5~10 are test patterns. We would like to check here the robustness property of the neural network with mosaic pattern. Many simulation examples give the result that both the tolerance degrees up-down and left-right rotation of the face are between  $\pm 15^\circ$ . If we take degree much bigger then recognition error

rate is getting higher abruptly. This means that the fundamental feature of the face changes strongly depending upon its degree over  $\pm 15^\circ$ . Within  $\pm 15^\circ$  we got 100% recognition. Hence our simulation shows the mosaic pattern is tolerant for face rotation, difference of the face expressions, and the image pixel shift.



Fig.1  
original front  
face



Fig.2  
mosaic pattern  
of Fig.1



Fig.3  
left rotate  
face



Fig.4  
mosaic pattern  
of Fig.3



Fig.5  
right rotate  
face



Fig.6  
mosaic pattern  
of Fig.5



Fig.7  
up-facing face

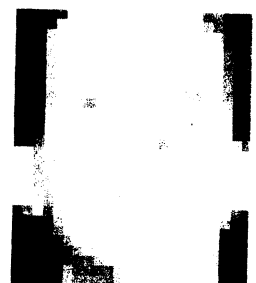


Fig.8  
mosaic pattern  
of Fig.7



Fig.9  
down-facing  
face

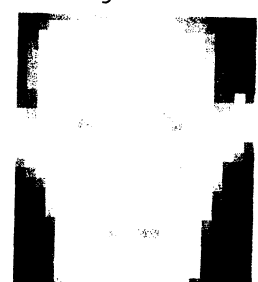


Fig.10  
mosaic pattern  
of Fig.9

#### 4. Conclusion

We have examined the tolerance property of the neural network with a mosaic pattern. The simulation examples show that the presented neural networks work so well for a face-swinging between  $\pm 15^\circ$ . We set the corresponding output unit value 0.9 and the other ones 0.1 for repeating study of the neural network. Future problem is in the number of the object persons. Now we have set it 20 persons. In real world, however, more than 100 persons should be applicable to such a system. Hence we will try to make a bigger system with 100 output neurons. If the recognition for more

than one hundred persons is required, then we can make many of the systems for 100 persons run in parallel. And finally, in the case which we want to recognize the face swinging more than 15 degrees it should be needed to make such a teacher's pattern. It will become getting important in near future more and more to build such a system up a security branch, robotics, and various other fields.

#### References

- (1)Sirovich L, Kirby M (1987), Low-dimensional procedure for the characterization of human faces. J. Opt. Soc. Am. A, 4, 3, pp.519-524
- (2)Turk M, Pentland A (1991), Face recognition using eigengfaces. Proc. CVPR'91
- (3)Harmon LD, Julesz B (1973) Masking in visual recognition: Effects of two-dimensional filtered noise. Science, 180, 15, pp.1194-1197
- (4)Makoto K (1993) Human-face recognition using mosaic pattern and neural networks (in Japanese) Trans. of IEICE, D-II, No.6, pp.1132-1139

# Diagnosis With Fuzzy Belief Networks

Indrani Chakraborty

Dept. of Elect. & Ecs. Engg., Mie University, Mie 514-8507, Japan.

Amit Konar and Ajit K. Mandal

Dept. of Ecs. & Tele-Comm. Engg., Jadavpur University, Calcutta 700 032, India.

*Abstract: The state space approach for belief computation by fuzzy means is a major work in the paper. The diagnostic problem thus is casted in the flavor of the state-space formulation of the belief networks. The most noticeable feature of the paper is the formulation of a new kind of inverse fuzzy relation with respect to AND-OR composition operation of fuzzy matrices. A numerical algorithm for computing fuzzy inverse matrix has been presented and its convergence has been proved. The algorithm for inverse fuzzy relation has been subsequently used for backward reasoning in a casual network for diagnosis of electrical appliances.*

## I. INTRODUCTION:

The paper presents a new scheme of diagnosis by employing fuzzy belief networks. The scheme is analogous to belief reunion mechanism of Judea Pearl applied to evidential network. When stochastic beliefs are not enumerable, one may use this scheme.

In late 80's Prof. Judea Pearl of University of California enunciated a new concept of belief networks [7] using Bayesian statistics. His pioneering contribution was then enhanced and applied in many areas of Engg., especially in diagnosis problems. Such networks, however, presumes a number of conditional probabilities, which are very difficult to extract for many complex problems of the real world.

Prof. Glenn Shafer too extended the fundamental theory of estimating orthogonal summation of beliefs of evidences. The theory, now popularly called the Dempster-Shafer theory. This theory has been applied in avoiding ambiguity in evidences, collected from multiple non-authentic sources. Prof. Kak for instance, applied Dempster-Shafer theory in recognizing objects from their multiple images. Our serious limitation of the Dempster-Shafer theory is the requirement of polynomial time for executing orthogonal summation of beliefs collected from  $N$  evidences. Suffer and Logan [8] devised a simplified method for estimation of composite belief in a linear time. However, they lost much accuracy in order to produce computational complexity. The paper to the best of the authors' beliefs is the first theoretical work on belief computation employing the Fuzzy logic. The logic has already proven its success in many complex

problems surmounted with inexactness of data and knowledge. It has also been applied in diagnosing faults in physical system. The approximation (or vagueness) of both data knowledge will be reduced with more information from multiple sources by employing his logic. This structural scheme for propagation of beliefs from one node to another in an evidential space is kept more or less similar with Pearl's paradigms. The principle of Fuzzy orthogonal summation presented in the paper, would also be used for fusing data/knowledge from multiple sources like Dempster-Shafer theory.

The paper presents a new method for computing belief in a fuzzy acyclic casual network, that represent the cause-effect relationship among a set of events. The nodes at the input layer of the network corresponds to the defects, while the nodes at the output layer corresponds to the measurements points of the diagnosis problem. The principle of the data fusion in Bayesian stochastic calculus has been extended here using fuzzy implication functions. The membership distribution at the measurement points are submitted one by one and a backward reasoning algorithm is invoked to compute the distribution of all nodes up to the starting layer, hereafter called axioms, in the network. Once such propagation of membership distribution up to the axioms is over, a forward reasoning algorithm is called for. The forward reasoning algorithm estimates the layer wise distribution up to the last (output) layer. The process of propagation from the last to the first layer is carried out by backward reasoning, and the propagation of membership towards the last layer from the

axioms is then activated in the second pass. The process continues for n passes until the belief at a fixed node in the output layer attains a maximum value, which remains practically invariant in the subsequent passes. This is an equilibrium condition in the network. The axioms with the highest belief in the equilibrium condition is then selected as the defect. The proposed scheme is better than the usual Bayesian reasoning for it does not demand any conditional probabilities.

Section-II in the paper deals with the structural organization of the evidential space. In section-III a details scheme for belief propagation will be presented. The scheme for belief propagation will be applied in diagnosis problem in section-IV. Concluding remarks arrived out of the paper will be briefly narrated in section-V.

## II. THE ORGANIZATION OF THE EVIDENTIAL SPACE:

The evidential space in this paper consists of a set of rules, whose certainty is not guaranteed. It also includes a set of imprecized data. For instance let us consider the following rules and a data based that supports the antecedents of the rules.

Rulebase rule1    A, B  $\rightarrow$  C, D  
                           C  $\rightarrow$  E, F  
                           F  $\rightarrow$  G  
 Supplied database    A, B

A structure representing the evidential space for the system can be presented (vide Fig.1)

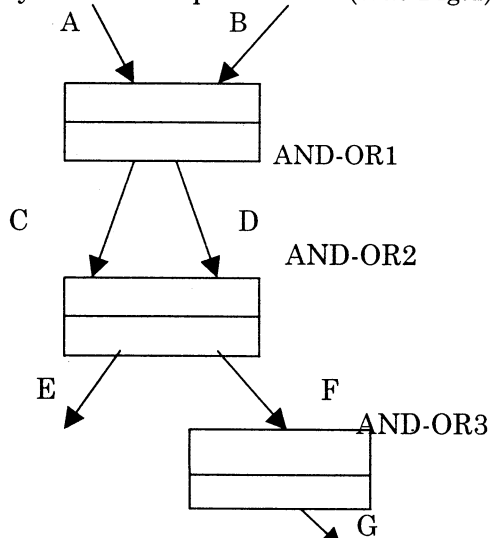


Fig 1. :The evidential space for the above rule and database

Now since the rules are not certain and the data too are imprecise we attach a Fuzzy certainty factor with each rule and a Fuzzy token distribution to each node (even) in the graph. Let the membership distribution for A,B,C,D,E,F,G be denoted by vectors like  $\mu_A, \mu_B, \mu_C, \mu_D, \mu_E, \mu_F, \mu_G$  respectively. The Fuzzy certainty factor [1-5], are also attached with each and/or module. For example if the certainty factor of the rule A, B  $\rightarrow$  C, D corresponding to and/or module1 be vector  $\mu_1$ . Similarly, the certainty associated with AND-OR logic j be  $\mu_j$ , for all  $j \leq 3$ .

Given the fuzzy belief distribution  $\mu_A$  and  $\mu_B$ , one can compute the belief distribution of G by the following equations.

$$\begin{aligned}\mu_C &= (\mu_A \wedge \mu_B) \wedge \mu_1 \\ \mu_F &= \mu_C \wedge \mu_2 \\ \mu_G &= \mu_F \wedge \mu_3\end{aligned}$$

We now represent the evidential structure by a vector-matrix approach. To denote the notion of time we add one parameter t, with each vector. Further, the AND of two vectors is identical with complement of the OR of complements of individual vectors (DeMorgan's theorem).

Formally,  $\mu_A \wedge \mu_B = (\mu_A^C \vee \mu_B^C)^C$  where C denotes the complementation

operation. The  $\wedge$  and  $\vee$  of two vectors are done component wise like conventional vector addition. The entire structure, now can thus be represented by the following equation.

$$\begin{aligned}\begin{pmatrix} \mu_C(t) \\ \mu_E(t) \\ \mu_G(t) \end{pmatrix} &= \begin{pmatrix} M_0 & \begin{pmatrix} \mu_A(t) \\ \mu_B(t) \\ \mu_G(t) \end{pmatrix}^C \end{pmatrix}^C \wedge \begin{pmatrix} \mu_1 \\ \mu_2 \\ \mu_3 \end{pmatrix} \\ \Rightarrow \mu_0 &= (M_0 \mu_1^C)^C \wedge \mu_{CF}\end{aligned}$$

Where M is a incidence matrix of the following form.

	A	B	C	D	E	F	G
AND-OR1	1	1	0	0	0	0	0
AND-OR2	0	0	1	0	0	0	0
AND-OR3	0	0	0	0	0	1	0

### III. BACKWARD REASONING WITH BELIEF NETWORK:

$\mu_0$ , the output belief vector of the system, thus can be estimated by above expression. Now one can evaluate  $(M_0 \mu_1^C)^C$  by employing

$$(M_0 \mu_1^C)^C = \mu_0 \cup [\mu_{CF} \cdot \mu_0] \vee \mu_{CF} \cup [\mu_0 \cdot \mu_{CF}] \\ = \mu' \quad (\text{say})$$

$$\Rightarrow \mu_1 = [M_0^{-1} (\mu')^C]^C \quad \text{.....(A)}$$

Where  $-1$  is power of  $M$  denotes fuzzy inverse with respect to AND-OR composition operator.

Estimation of fuzzy inverse matrix :

It has been observed that expression (A) that for evaluation of  $\mu_1$ , one requires to estimate  $M^{-1}$ , where  $-1$  is power of  $M$  denotes the inverse of  $M$  with respect to AND-OR composition operator. This section will be denoted to present a scheme for evaluation of the AND-OR compositional inverse of the fuzzy matrix  $M$ . To demonstrate the scheme for computation of  $M^{-1}$ , we first illustrate it with a  $(3 \times 3)$  matrix. The generalisation of the scheme, however is efficient from the example. Let

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

be a fuzzy matrix ,

i.e. Elements  $0 \leq a_{ij} \leq 1 \quad \forall \quad i, j$

Now we want  $A_0 A^{-1} \rightarrow I$

i.e. the product of  $A$  by its inverse should be sufficiently close to  $I$ , the identity matrix. In fact  $A_0 A^{-1} = I$  when  $A = A^{-1} = I$ .

For no other  $A (\neq I)$ ,  $A_0 A^{-1} \neq I$

but  $A_0 A^{-1} \rightarrow I$

We will solve the problem of computing  $A^{-1}$  by using a set of constraints which will be determined by the concept presented below,

Let  $A_0 A^{-1} \rightarrow I$  i.e.

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Where  $b_{ij}$  denotes the  $(i, j)$  th element of  $A^{-1}$ . Now, for evaluation of  $b_{ij}$ , we would first check which are the operands in  $A$  that operates with  $b_{ij}$ . For instance let us try to evaluate  $b_{31}$ . It may be noted that  $b_{31}$  can operate with  $a_{13}$ ,  $a_{23}$  and  $a_{33}$  only. When  $b_{31}$  operates with  $a_{13}$ , the ANDed product should

be maximized to get a result closest possible to  $I_{(1,1)} = 1$  i.e.  $(b_{31} \wedge a_{13})$  is to be maximized.

----- (B)

Secondly  $(b_{31} \wedge a_{23})$  is to be minimized so that the result approaches

$I_{(2,1)} = 0$  ----- (C).

Thirdly,  $(b_{31} \wedge a_{33})$  is to be minimized so that the ANDed product approaches

$I_{(3,1)} = 0$ . ----- (D)

Combining (B) (C) (D), we may construct a new criteria given by maximize  $Z$ , where  $Z = (b_{31} \wedge a_{13}) - \{(b_{31} \wedge a_{23}) + (b_{31} \wedge a_{33})\}$

Now one may vary  $b_{31}$  in the closed +interval  $[0,1]$  and check for which value of  $b_{31}$ ,  $Z$  is maximized. This, however calls for a search in a wide range of interval. One way to minimize the search cost is to reduce the domain of search. It has been proved [6] that one does not loose any value of  $Z$  if the search domain is set as

$$[0, \{ \min(a_{13}, a_{23}, a_{33}), \max(a_{13}, a_{23}, a_{33}) \}.]$$

The idea of estimating inverse for a  $(n \times n)$  matrix thus can be easily envisaged following the example discussed in Section-IV.

### IV. DIAGNOSIS BY FUSING INFORMATION FROM MULTIPLE SOURCES:

It has already been narrated that the casual relationship in general can be represented by a tree, where the parent denotes the cause and the children are the effects. For instance, consider the following tree representing the casual relationship among evidences describing why a car does not start.

Now to determine the possible cause of the defective system we here use the well known concept of stochastic belief propagation following Judea pearl [7]. In Pearl's scheme for evidential reasoning he measured the believe at any of the leaves artificially and then propagated of up to the root of the tree and then gradually down the tree until the beliefs reach the leaves. Now among the competitive belief of the terminal nodes leaves the node which the highest belief is identified and the measured belief for that node is fused into the system by the following scheme.

Let  $E$  be the node with highest measured belief. The propagation from  $E$  then continues following the scheme presented in Fig.2, where the arrowhead

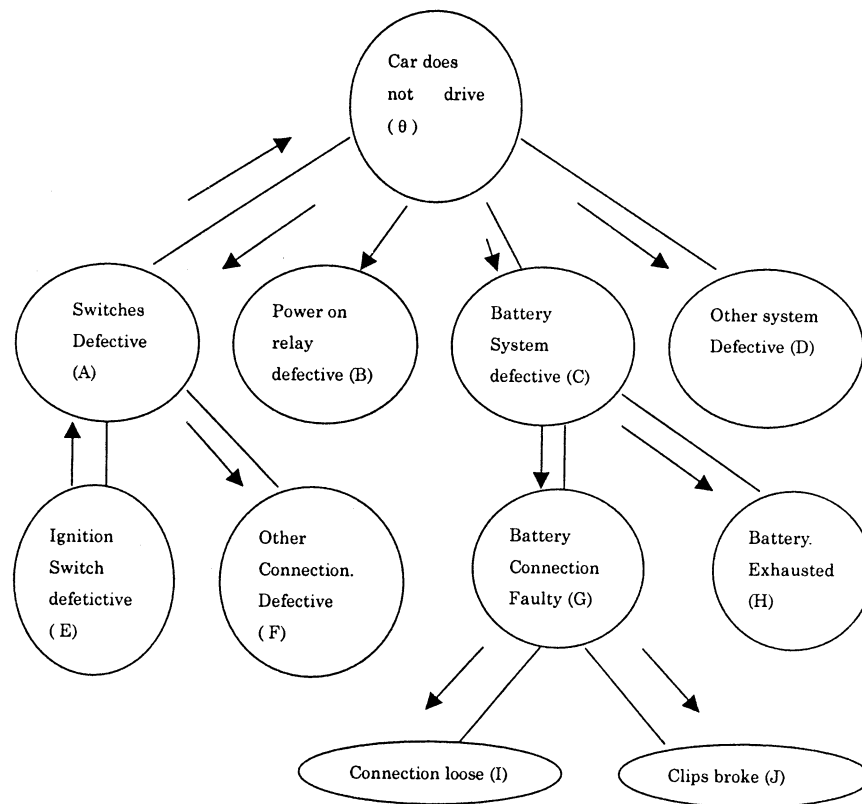


Fig.2 A causal tree describing the possible causes of failure in start-up of a car.

( $\rightarrow$ ) denotes the direction of belief movement. When belief move up the tree one should use backward reasoning, on the other hand when the beliefs move down the tree, one should use forward reasoning. The process is thus constituted many many times unless a situation occurs when belief of the leaf node remaining the highest among all its competitive partner leaves. The leaf node which thus demonstrates to progress the highest belief is selected as the cause for the system to be defective.

## V. CONCLUSIONS:

Pearls's scheme for belief reunion is extended in this paper with Fuzzy models. The bi-directional scheme for the belief movement on an evidential graph have been developed using inverse Fuzzy relation. An idea of estimating Fuzzy inverse by AND-OR composition operator is illustrated for application in diagnosis problem. The proposed scheme is highly time efficient and thus most appropriate for real time realization.

## REFERENCES:

[1] A.Konar and A.K.Mandal, "Uncertainty management in Expert systems

using Fuzzy petrinets", IEEE Trans. on Knowledge & data Engg. Feb. 1996.

[2] S.Pal and A.Konar, "Cognitive reasoning with Fuzzy Neural nets", IEEE Trans. on Sys. Man & Cyber., August 1996.

[3] A.Konar, "Artificial Intelligence and Soft Computing: behavior and cognitive modeling of the human brain" (in Press) CRC USA.

[4] A.Konar and S.Pal, "Modelling Cognition with Fuzzy neural nets in Fuzzy logic system: Techniques and Applications", Ed. C.T.Leondes, Academic press (to appear).

[5] B.Pal, A.Konar and A.K.Mandal, "Fuzzy ADALINES for gray image recognition " Neuro computing, Elsevier, North holland, to appear.

[6] P.Saha and A.Konar " Backward reasoning with inverse fuzzy relations," Proc. Of Inter. Conf. On control, Automation, Robotics and computer vision, 1996.

[7] J.Pearl, "Distributed Reunion of Composite beliefs," Artificial Intelligence, Vol.29, 1996.

[8] G.Shafer and R.Logan, "Implementing Dempster's rule for hierarchical evidence," Artificial Intelligence, Vol. 33 1997.

# Design of Output Feedback Controllers for Takagi-Sugeno Fuzzy Descriptor Systems

Jun Yoneyama, Kenji Itoh and Akira Ichikawa  
Department of Electrical and Electronic Engineering,  
Faculty of Engineering, Shizuoka University  
5-1 Johoku 3-chome, Hamamatsu, 432-8561 Japan  
Phone/Fax: +81-53-478-1132  
Email: yoneyama@eng.shizuoka.ac.jp

## Abstract

In this paper the design of output feedback controllers for Takagi-Sugeno fuzzy descriptor models is considered. First fuzzy descriptor models are introduced. Then we consider state feedback controllers and observers for such models. We then show that a state feedback controller and an observer always yield a stabilizing output feedback controller provided that the stabilizing property of the control and the asymptotic convergence of the observer are guaranteed by the Lyapunov method. In this sense it is shown that the separation principle holds for Takagi-Sugeno fuzzy descriptor systems.

### Keywords

Descriptor Systems; Takagi-Sugeno Fuzzy Models;  
Output Feedback Stabilization

## 1 Introduction

Takagi-Sugeno fuzzy models (Takagi and Sugeno<sup>3</sup>, Tanaka and Sano<sup>6</sup>) are nonlinear systems described by a set of if-then rules which gives a local linear representation of an underlining system. Such models can approximate a wide class of nonlinear systems. They can even describe exactly certain nonlinear systems in Kawamoto et al<sup>1</sup>. and Tanaka et al<sup>5</sup>. Hence it is important to study their stability or the synthesis of stabilizing controllers. The output stabilization problem for fuzzy systems has been considered in Yoneyama et al<sup>8,9</sup>, and it has been shown that the separation principle holds for the fuzzy systems there.

In this paper we extend the results in Yoneyama<sup>8,9</sup> to a class of fuzzy descriptor systems. We first give the sufficient conditions for the admissibility of state feedback controllers. Then we introduce a natural form of observers. We finally consider the stabilization problem by the output feedback based on the observer.

## 2 The Fuzzy Descriptor Model

In this section, we introduce fuzzy descriptor models of Takagi and Sugeno and the admissibility conditions for the homogeneous models.

Consider the fuzzy descriptor model of Takagi and Sugeno described by the following fuzzy IF-THEN rules;

$$\begin{array}{ll} \text{IF} & z_1 \text{ is } M_{1i} \text{ and } \cdots \text{ and } z_p \text{ is } M_{pi}, \\ \text{THEN} & E\dot{x}(t) = A_i x(t) + B_i u(t), \\ & y(t) = C_i x(t), \quad i = 1, \dots, r, \end{array} \quad (1)$$

where  $x(t) \in \mathbb{R}^n$  is the descriptor variable,  $u(t) \in \mathbb{R}^m$  is the control input,  $y(t) \in \mathbb{R}^q$  is the output. The matrices  $E$ ,  $A_i$ ,  $B_i$  and  $C_i$  are of appropriate dimension, and  $E_i$  may not have full rank.  $r$  is the number of IF-THEN rules, and  $M_{ij}$  are the fuzzy sets.  $z_1, \dots, z_p$  are premise variables. We set  $z = [z_1 \ z_2 \ \cdots \ z_p]^T$ . It is assumed that the premise variables do not depend on control variables.

Then the state equation and the output are defined as follows;

$$E\dot{x}(t) = \sum_{i=1}^r \lambda_i(z(t))(A_i x(t) + B_i u(t)), \quad (2)$$

$$y(t) = \sum_{i=1}^r \lambda_i(z(t))C_i x(t), \quad (3)$$

where

$$\lambda_i(z(t)) = \frac{w_i(z(t))}{\sum_{i=1}^r w_i(z(t))}, \quad w_i(z(t)) = \prod_{j=1}^p M_{ij}(z_j),$$

and  $M_{ij}(\cdot)$  are the membership functions of the fuzzy sets of  $M_{ij}$ . We assume

$$w_i(z(t)) \geq 0, \quad i = 1, \dots, r, \quad \sum_{i=1}^r w_i(z(t)) > 0 \quad \forall t.$$

Hence  $\lambda_i(z(t))$  satisfy

$$\lambda_i(z(t)) \geq 0, \quad i = 1, \dots, r, \quad \sum_{i=1}^r \lambda_i(z(t)) = 1 \quad \forall t.$$

First consider a descriptor system

$$\begin{array}{ll} E\dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t). \end{array} \quad (4)$$



**Definition 2.1** (i) A pencil  $sE - A$  (or a pair  $(E, A)$ ) is regular if  $\det(sE - A)$  is not identically zero.

(ii) For a regular pencil  $sE - A$ , the finite eigenvalues of  $sE - A$  are said to be the finite modes of  $(E, A)$ . Suppose that  $Ev_1 = 0$ . Then the infinite eigenvalues associated with the generalized principle vectors  $v_k$  satisfying  $Ev_k = Av_{k-1}$ ,  $k = 2, 3, \dots$ , are impulsive modes of  $(E, A)$ .

(iii) A pair  $(E, A)$  is admissible if it is regular and has neither impulse

For the descriptor system (4) with  $u(t) = 0$ , the following admissibility condition is known (Masubuchi<sup>2</sup>).

**Lemma 2.1** A pair  $(E, A)$  is admissible if and only if there exists a common matrix  $X$  such that

$$\begin{aligned} E^T X &= X^T E \geq 0, \\ A^T X + X^T A &< 0. \end{aligned}$$

When we consider the stabilization of (2), it is useful to have admissibility conditions for the homogeneous system (5);

$$E\dot{x}(t) = \sum_{i=1}^r \lambda_i(z(t)) A_i x(t). \quad (5)$$

If pairs  $(E, A_i)$  are admissible, the following conditions for the admissibility can be derived.

**Theorem 2.1** The fuzzy system (5) has a solution and is exponentially stable if there exists a common matrix  $X$  such that

$$\begin{aligned} E^T X &= X^T E \geq 0, \\ A_i^T X + X^T A_i &= -P_i \leq -aI, \quad i = 1, \dots, r \end{aligned} \quad (6)$$

where  $P_i > 0$ .

In order to prove the theorem, we need the following lemma.

**Lemma 2.2** Given a function  $A(t)$  of  $t$ . Suppose that there exists a matrix  $X$  such that

$$A^T(t)X + X^T A(t) \leq -\alpha I \quad (7)$$

for all  $t$  and some positive number  $\alpha$ . Then the followings hold.

(i)  $A(t)$  is invertible,

(ii)  $|A^{-1}(t)| \leq aI$  for some finite number  $a$ .

**Proof:** Suppose (7) holds. Then it is easy to see that  $A(t)$  is nonsingular. Hence (i) holds. For (ii), let  $y$  be given and let  $x = A^{-1}(t)y$ . Then it follows from (7) that

$$\begin{aligned} 2 < A(t)x, Xx > &\leq -\alpha|x|^2, \\ \frac{2}{\alpha} < y, XA^{-1}(t)y > &\leq -|A^{-1}(t)y|^2, \\ \frac{\alpha}{2} |A^{-1}(t)y|^2 &\leq -y^T X A^{-1}(t)y \\ &\leq |X^T y| |A^{-1}(t)y|, \\ |A^{-1}(t)y| |A^{-1}(t)y| - \frac{2}{\alpha} |X^T y| &\leq 0, \\ 0 \leq |A^{-1}(t)y| &\leq \frac{2}{\alpha} |X^T y|, \\ &\leq \frac{2}{\alpha} |X^T| |y|, \\ &\leq a|y| \end{aligned}$$

where  $a = \frac{2}{\alpha} |X^T|$ . This proves (ii).  $\square$

**Proof of Theorem 2.1:** Without loss of generality, we assume that the system (5) is in the SVD coordinate where

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad E = \begin{bmatrix} \Sigma & 0 \\ 0 & 0 \end{bmatrix}, \quad A_i = \begin{bmatrix} A_{11}^i & A_{12}^i \\ A_{21}^i & A_{22}^i \end{bmatrix},$$

Suppose (6) holds. For the system (5), we have

$$\begin{aligned} &(\sum_{i=1}^r \lambda_i(z(t)) A_i)^T X + X (\sum_{i=1}^r \lambda_i(z(t)) A_i) \\ &= \begin{bmatrix} \Delta_{11} & \Delta_{12} \\ \Delta_{12}^T & \Delta_{22} \end{bmatrix} < 0 \end{aligned}$$

where

$$\begin{aligned} \Delta_{11} &= \mathcal{A}_{11}^T X_{11} + X_{11}^T \mathcal{A}_{11} + \mathcal{A}_{21}^T X_{21} + X_{21}^T \mathcal{A}_{21}, \\ \Delta_{12} &= \mathcal{A}_{21}^T X_{22} + X_{11}^T \mathcal{A}_{12} + X_{21}^T \mathcal{A}_{22}, \\ \Delta_{22} &= \mathcal{A}_{22}^T X_{22} + X_{22}^T \mathcal{A}_{22}. \end{aligned}$$

and we denote  $\sum_{i=1}^r \lambda_i(z(t)) A_i^k$  by  $\mathcal{A}_k$ . It follows that  $\Delta_{22} > 0$ , and hence  $\mathcal{A}_{22}$  is invertible by Lemma 2.2. Thus,  $x_2(t)$  can be expressed as

$$x_2(t) = -\mathcal{A}_{22}^{-1} \mathcal{A}_{21} x_1(t). \quad (8)$$

Substitution of (8), we obtain

$$\Sigma \dot{x}_1(t) = (\mathcal{A}_{11} - \mathcal{A}_{12} \mathcal{A}_{22}^{-1} \mathcal{A}_{21}) x_1(t)$$

where we denote  $\sum_{i=1}^r \lambda_i(z(t)) A_i^k$  by  $\mathcal{A}_k$ . Consider a function  $V(x, t) = x_1^T \Sigma^T X_{11} x_1$  where  $X_{11}$  is  $(1, 1)$ -block of  $X$ , and differentiate it along the solution of (5);

$$\begin{aligned} \dot{V}(x, t) &= x_1^T \Sigma^T X_{11} \dot{x}_1 \\ &= x_1^T E^T X \dot{x} \\ &= \sum_{i=1}^r \lambda_i(z(t)) x_1^T [A_i^T X + X^T A_i] x \\ &\leq -a|x|^2 \\ &\leq -a|x_1|^2. \end{aligned}$$

This proves the exponential stability of  $x_1(t)$ , and hence  $x_2(t)$  because of (8).  $\square$

**Remark 2.1** The fuzzy system (5) does not necessarily has a solution and is not exponentially stable even if all  $(E, A_i)$  are admissible.

### 3 Output Stabilization

In this section, we show the output stabilization of fuzzy descriptor systems when the premise variables are given function.

We assume that the state is fully available, i.e., all  $C_i$  are identity matrices. For each subsystem in (1), we assume that the following rules are given;

$$\begin{aligned} \text{IF} \quad & z_1 \text{ is } M_{1i} \text{ and } \dots \text{ and } z_p \text{ is } M_{pi}, \\ \text{THEN} \quad & u(t) = F_i x(t), \quad i = 1, \dots, r. \end{aligned}$$

Then the natural choice of the controller is the following (Tanaka<sup>4</sup>, Tanaka and Sugeno<sup>7</sup>);

$$u(t) = \sum_{i=1}^r \lambda_i(z(t)) F_i x(t). \quad (9)$$

We use the same weights  $\lambda_i(z(t))$  as those for the rules (1) of the fuzzy descriptor system. Substituting the controller (9) into the fuzzy descriptor system (2), we obtain

$$E\dot{x}(t) = \sum_{i=1}^r \sum_{j=1}^r \lambda_i(z(t))\lambda_j(z(t))(A_i + B_i F_j)x(t). \quad (10)$$

The above equation can also be written as

$$E\dot{x}(t) = \sum_{i=1}^r \lambda_i^2(z(t))(A_i + B_i F_i)x(t) + \sum_{i < j} 2\lambda_i(z)\lambda_j(z) \frac{\{(A_i + B_i F_j) + (A_j + B_j F_i)\}}{2} x(t). \quad (11)$$

Even if all  $(E, A_i + B_i F_j)$  are admissible, it does not guarantee the unique solution and the exponential stability of the closed-loop system (10). But the direct application of Theorem 2.1 yields the following sufficient conditions for the unique solution and exponential stability of (10).

**Theorem 3.1** (i) *The fuzzy system (10) has a solution and is exponentially stable if there exists a common matrix  $X$  such that*

$$\begin{aligned} E^T X &= X^T E \geq 0, \\ (A_i + B_i F_j)^T X + X^T (A_i + B_i F_j) &= -P_{ij}, \quad \forall i, j, \end{aligned} \quad (12)$$

where  $P_{ij} > 0$ .

(ii) *The fuzzy system (11) (and hence (10)) has a solution and is exponentially stable if there exists a common matrix  $X$  such that*

$$\begin{aligned} E^T X &= X^T E \geq 0, \\ (A_i + B_i F_i)^T X + X^T (A_i + B_i F_i) &= -P_i, \quad \forall i, \\ G_{ij}^T X + X^T G_{ij} &= -\frac{1}{2} P_{ij}, \quad i < j, \end{aligned} \quad (13)$$

where

$$G_{ij} = \frac{1}{2}[(A_i + B_i F_j) + (A_j + B_j F_i)],$$

and  $P_i > 0$  and  $P_{ij} > 0$ .

The condition (ii) is in general weaker than the condition (i).

Now we consider for (2) an observer form

$$E\dot{\hat{x}}(t) = \sum_{i=1}^r \lambda_i(z(t))(A_i \hat{x}(t) + B_i u(t)) + K(y(t) - \hat{y}(t)), \quad (14)$$

where  $\hat{y}(t)$  is given by

$$\hat{y}(t) = \sum_{i=1}^r \lambda_i(z(t))C_i \hat{x}(t).$$

We wish to find  $K$  such that  $e(t) = x(t) - \hat{x}(t) \rightarrow 0$  exponentially as  $t \rightarrow \infty$ . We assume that the following rules are given concerning the observer of each subsystem in (1);

IF  $z_1$  is  $M_{1i}$  and  $\dots$  and  $z_p$  is  $M_{pi}$ ,  
THEN  $E\dot{\hat{x}}(t) = A_i \hat{x}(t) + B_i u(t) + K_i(y(t) - \hat{y}(t))$ ,  $i = 1, \dots, r$ .

We propose the following observer gain;

$$K = \sum_{i=1}^r \lambda_i(z(t))K_i. \quad (15)$$

Substituting (15) into (14), we have

$$E\dot{\hat{x}}(t) = \sum_{i=1}^r \lambda_i(z(t))(A_i \hat{x}(t) + B_i u(t)) + \sum_{j=1}^r \lambda_j(z(t))K_j(y(t) - \hat{y}(t)). \quad (16)$$

Subtracting (16) from (2), we have the error system

$$E\dot{e}(t) = \sum_{i=1}^r \sum_{j=1}^r \lambda_i(z(t))\lambda_j(z(t))(A_i - K_j C_i)e(t). \quad (17)$$

The above equation can also be written as

$$E\dot{e}(t) = \sum_{i=1}^r \lambda_i^2(z(t))(A_i - K_i C_i)e(t) + \sum_{i < j} 2\lambda_i(z)\lambda_j(z) \frac{\{(A_i - K_j C_i) + (A_j - K_i C_j)\}}{2} e(t). \quad (18)$$

The systems (17) and (18) have the dual forms to the systems (10) and (11), respectively. Hence the sufficient conditions for  $e(t) \rightarrow 0$  exponentially as  $t \rightarrow \infty$  can be found using Theorem 2.1.

**Theorem 3.2** (i) *The error system (17) has a solution and is exponentially stable if there exists a common matrix  $Y$  such that*

$$\begin{aligned} E^T Y &= Y^T E \geq 0, \\ (A_i - K_j C_i)^T Y + Y^T (A_i - K_j C_i) &= -Q_{ij}, \quad \forall i, j, \end{aligned} \quad (19)$$

where  $Q_{ij} > 0$ .

(ii) *The error system (18) (and hence (17)) has a solution and is exponentially stable if there exists a common matrix  $Y$  such that*

$$\begin{aligned} E^T Y &= Y^T E \geq 0, \\ (A_i - K_i C_i)^T Y + Y^T (A_i - K_i C_i) &= -Q_i, \quad \forall i, \\ L_{ij}^T Y + Y^T L_{ij} &= -\frac{1}{2} Q_{ij}, \quad i < j, \end{aligned} \quad (20)$$

where

$$L_{ij} = \frac{1}{2}[(A_i - K_j C_i) + (A_j - K_i C_j)],$$

and  $Q_i > 0$  and  $Q_{ij} > 0$ .

It is well-known that for linear systems the combination of a stabilizing state feedback and an observer yields a stabilizing output feedback controller. This is known as the separation principle. It has been shown in Yoneyama et al<sup>8,9</sup> that the separation principle holds for fuzzy systems. Here we extend the results to fuzzy descriptor systems. Consider the system (2), the observer given by (16) and

$$u(t) = \sum_{i=1}^r \lambda_i(z(t))F_i \hat{x}(t). \quad (21)$$

Combining (2), (3), (17) and (21), we have

$$\tilde{E}\dot{\tilde{x}}(t) = \sum_{i=1}^r \sum_{j=1}^r \lambda_i(z(t))\lambda_j(z(t))H_{ij}\tilde{x}(t), \quad (22)$$

where  $\tilde{x} = [x^T \ e^T]^T$  and

$$\tilde{E} = \begin{bmatrix} E & 0 \\ 0 & E \end{bmatrix}, \quad H_{ij} = \begin{bmatrix} A_i + B_i F_j & -B_i F_j \\ 0 & A_i - K_j C_i \end{bmatrix}.$$

The above equation can also be written as

$$\begin{aligned} \tilde{E} \dot{\tilde{x}}(t) &= \sum_{i=1}^r \lambda_i^2(z(t)) H_{ii} \tilde{x}(t) \\ &+ \sum_{i < j}^r 2\lambda_i(z(t)) \lambda_j(z(t)) \frac{(H_{ij} + H_{ji})}{2} \tilde{x}(t). \end{aligned} \quad (23)$$

Since (22) is of the form (5), we can apply Theorem 2. 1 to (22). An obvious candidate of the common matrix in Theorem 2. 1 is  $\begin{bmatrix} X & 0 \\ 0 & Y \end{bmatrix}$  if there exist common matrices  $X$  and  $Y$  satisfying the conditions of Theorems 3. 1 and 3. 2. However, the matrix above does not in general satisfy the conditions of Theorem 2. 1 (see Tanaka and Sano<sup>6</sup>). We shall show that

$$\bar{X} = \begin{bmatrix} X & 0 \\ 0 & \alpha Y \end{bmatrix} \quad (24)$$

for sufficiently large  $\alpha > 0$  guarantees that (22) has a solution and is exponentially stable. Hence, if we can design a state feedback controller and an observer as in Section 3, we can always construct a stabilizing output feedback controller.

To prove the admissibility of (22), consider the controller given by (16) and (21) and suppose that there exist  $X > 0$  and  $Y > 0$  satisfying (12) and (19), respectively. Then for  $\bar{X}$  in (24), we have

$$\tilde{E}^T \bar{X} = \bar{X}^T \tilde{E} \geq 0, \quad (25)$$

$$H_{ij}^T \bar{X} + \bar{X}^T H_{ij} = - \begin{bmatrix} P_{ij} & X^T B_i F_j \\ F_j^T B_i^T X & \alpha Q_{ij} \end{bmatrix}. \quad (26)$$

By Shur complement, the necessary and sufficient condition for the right-hand-side of the equation (26) to be negative definite is

$$\alpha Q_{ij} > F_j^T B_i^T X (P_{ij})^{-1} X^T B_i F_j, \quad \forall i, j = 1, \dots, r. \quad (27)$$

The quantities on the right-hand-side of (27) depend only on the fuzzy state feedback controller, while  $Q_{ij}$  on the left-hand-side depend only on the fuzzy observer. Hence we can always choose  $\alpha$  large enough so that (27) hold. Thus by Theorem 2. 1, we have the following result.

**Theorem 3. 3** *Suppose that there exist  $X > 0$  and  $Y > 0$  that satisfy (12) and (19), respectively. Then we can always find a positive number  $\alpha$  such that (26) are negative definite. Moreover, the fuzzy system (22) has a solution and is exponentially stable.*

$$H_{ii}^T \bar{X} + \bar{X}^T H_{ii} = - \begin{bmatrix} P_i & X^T B_i F_i \\ F_i^T B_i^T X & \alpha Q_i \end{bmatrix}. \quad (28)$$

$$\begin{aligned} &\frac{(H_{ij} + H_{ji})^T}{2} \bar{X} + \bar{X}^T \frac{(H_{ij} + H_{ji})}{2} \\ &= -\frac{1}{2} \begin{bmatrix} P_{ij} & X^T (B_i F_j + B_j F_i) \\ (B_i F_j + B_j F_i)^T X & \alpha Q_{ij} \end{bmatrix}. \end{aligned} \quad (29)$$

**Theorem 3. 4** *Suppose that there exist  $X > 0$  and  $Y > 0$  that satisfy (13) and (20), respectively. Then we can always find a positive number  $\alpha$  such that (28) and (29) are negative definite. Moreover the fuzzy system (23) has a solution and is exponentially stable.*

## 4 Conclusion

We have considered the output stabilization problem of Takagi-Sugeno fuzzy models. First we have introduced a natural form of observers. Then we have had sufficient conditions for the stability of a state feedback and for the asymptotic convergence of an observer. Finally we have shown that a stabilizing state feedback combined by an asymptotically convergent observer satisfying the above conditions always yields a stabilizing output feedback controller and hence that the separation principle holds for Takagi-Sugeno fuzzy descriptor models.

## References

- [1] S. Kawamoto, K. Tada, N. Onoe, A. Ishigame and T. Taniguchi(1992), Construction of exact fuzzy system for nonlinear system and its stability analysis(in Japanese), 8th Fuzzy System Symposium, Hiroshima, Japan 517-520.
- [2] I. Masubuchi, Y. Kamitane, A. Ohara and N. Suda(1997),  $H_\infty$  Control for Descriptor Systems: A Matrix Inequalities Approach, Automatica 33:669-673.
- [3] 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.
- [4] K. Tanaka(1994), Advanced Fuzzy Control(in Japanese) Kyoritsu Publisher.
- [5] K. Tanaka, T. Ikeda and H. O. Wang(1996), Robust stabilization of a class of uncertain nonlinear systems via fuzzy control: quadratic stabilizability,  $H_\infty$  control theory, and linear matrix inequalities, IEEE Transaction on Fuzzy Systems 4:1-13.
- [6] K. Tanaka and M. Sano(1994), On the concepts of regulator and observer of fuzzy control systems, 3rd IEEE International Conference on Fuzzy Systems 767-772.
- [7] K. Tanaka and M. Sugeno(1992), Stability analysis and design of fuzzy control systems, Fuzzy Sets and Systems 45:135-156.
- [8] J. Yoneyama, M. Nishikawa, H. Katayama and A. Ichikawa(1998), Output stabilization of Takagi-Sugeno fuzzy systems, to appear in Fuzzy Sets and Systems.
- [9] J. Yoneyama, M. Nishikawa, H. Katayama and A. Ichikawa(1998), Design of output feedback controllers for Takagi-Sugeno fuzzy systems, submitted to Fuzzy Sets and Systems.

# Stability Study of Fuzzy Logic Control System for an Inverted Pendulum

Kazushi Nakano<sup>1</sup> and Masayoshi Tomizuka<sup>2</sup>

<sup>1</sup> Department of Electrical Engineering, Fukuoka Institute of Technology, 3-30-1 Wajiro-higashi,  
Higashi-ku, Fukuoka 811-02, JAPAN, E-mail:nakano@ee.fit.ac.jp

<sup>2</sup> Department of Mechanical Engineering, University of California at Berkeley,  
5100B Etcheverry Hall, Berkeley, CA 94720-1740, USA, E-mail:tomizuka@euler.me.berkeley.edu

## Abstract

The stability of a fuzzy logic control (FLC) system for an inverted pendulum is analyzed by the Lyapunov method. The FLC system is built with two cascaded fuzzy controllers. The stability of the pendulum with consideration of cart dynamics, is guaranteed by satisfying a set of inequalities, which include terms dependent on unknown parameters of the pendulum and the cart. Simulation and experimental results are presented to illustrate the robust stability of the proposed method.

**Keywords.** inverted pendulum, cascade architecture, fuzzy logic control, nonlinear control, robust stability, Lyapunov method

## 1. Introduction

The dynamics of inverted pendulum system is described by a set of nonlinear differential equations with unstable equilibrium states and singularities. Therefore, linear control laws are effective only in limited ranges. On the other hand, the inverted pendulum is a low order dynamic system, and it is not difficult for human beings to figure out appropriate control strategies for stabilizing about the unstable equilibrium state and for global motions such as swinging up. Thus, there have been many studies of developing fuzzy logic controllers for inverted pendulums<sup>1,2,3,4</sup>. For an inverted pendulum mounted on a moving cart, the regulation problem is more complicated if both the pendulum and the cart must be regulated because the problem becomes an SIMO (single-input/multi-output) problem<sup>5</sup>. In an example of the inverted pendulum mounted at the end of a rotating arm, stabilization of the pendulum at the upright position and regulation of the arm at the arbitrarily specified position can be achieved using two fuzzy rules for both the pendulum and the arm and an additional synthesizing fuzzy rule<sup>3</sup>. While the stability and robustness of the fuzzy logic control (FLC) law in ref.3 have been demonstrated by experiments, they have not been verified mathematically.

On the other hand, the sliding-mode fuzzy-logic control (SMFLC) has been proposed for a 2nd-order nonlinear system<sup>6</sup>. We need, however, some knowledge about the details of system dynamics.

We have already constructed a FLC system which consists of two cascade fuzzy controllers<sup>4,7</sup>. This approach is based on the following three steps. First, a FLC law of the pendulum is described in the form of *if ~ then* rules. Secondly, a FLC law of the cart is synthesized in the same manner. Thirdly, the dynamic influence of the cart on the pendulum is taken into account as the set point in the pendulum control system.

In this paper, we consider the stability of the pendulum control system by the Lyapunov direct method. The stability conditions for the system are obtained in terms of a set of inequalities, which include terms dependent on design parameters and unknown parameters of the system. The inequalities allow us to construct a control system, which remains robust in the presence of parameter uncertainties. The stability analysis should be considered as a local analysis using a linearized model. However, we can verify the robust stability of the proposed control system using simulations and experiments for an actual inverted pendulum.

## 2. Fuzzy Control of Inverted Pendulum

The proposed FLC is applicable to a general 2nd-order system with unknown parameters. We especially deal with the robust stabilization of an inverted pendulum as a class of nonlinear mechanical systems.

The motion equations of the inverted pendulum are described by

$$(M + m)\ddot{x} - ml \sin \theta \cdot \dot{\theta}^2 + ml \cos \theta \cdot \ddot{\theta} + D_x \dot{x} = G_0 u \quad (1)$$

$$ml \cos \theta \cdot \ddot{x} + (ml^2 + J)\ddot{\theta} - mgl \sin \theta + D_\theta \dot{\theta} = 0 \quad (2)$$

where all the parameters are not precisely known in their

values.

## 2.1. Control of Pendulum Angle

As the first step of control system design, we consider an inference law for controlling the pendulum angle.

The dynamic behavior of the pendulum angle is described on the  $\theta$ - $\dot{\theta}$  phase plane. Hereafter, we use  $\phi$  and  $r$  in polar coordinates as the variables of the condition part in *if ~ then* rule.

$$\phi = \tan^{-1}(k_2 \dot{\theta}) / (k_1 \theta) \quad (3)$$

$$r = \sqrt{(k_1 \theta)^2 + (k_2 \dot{\theta})^2} \quad (4)$$

where  $k_1$  and  $k_2$  are scaling factors. Based on these, a control input is written as

$$u = \text{Sat}(\phi) \cdot G(r) \quad (5)$$

$\text{Sat}(\phi)$  in (5) is a saturation function which maps  $\phi$  to  $[-1, 1]$ . When the pendulum is leaving from the upright equilibrium position,  $\theta$  and  $\dot{\theta}$  take the same sign. On the other hand, when moving toward the equilibrium position,  $\theta$  and  $\dot{\theta}$  take the opposite sign. We determine the sign of control input by the state  $(\theta, \dot{\theta})$  on the phase plane. This determination is made by the fuzzy inference. On the other hand,  $G(r)$  in (5) is a gain function which maps  $r$  to  $[0, u_m]$ . It is necessary to assign the absolute value of gain with a distance from the equilibrium point for improving the transient response.

## 2.2. Control of Pendulum and Cart

We next consider an inference law for the control of both the pendulum angle and the cart position. The input  $u$  is determined in the form

$$u = \text{Sat}(\phi_p) \cdot G(r_p) \quad (6)$$

$$\phi_p = \tan^{-1}(k_2 \dot{\theta}) / (k_1 \tilde{\theta}) \quad (7)$$

$$r_p = \sqrt{(k_1 \tilde{\theta})^2 + (k_2 \dot{\theta})^2} \quad (8)$$

where  $\phi_p$  and  $r_p$  in polar coordinates are used for the pendulum, and are used as the variables of the condition part in *if ~ then* rule.  $\text{Sat}(\phi_p)$  and  $G(r_p)$  in (6) are defined in the same manner as in (5).  $\tilde{\theta}$  is the virtual equilibrium point<sup>2,4</sup> to account for the motion of the cart, and is defined by

$$\tilde{\theta} = \theta + k_0 \theta_0 \quad (9)$$

where  $k_0$  is a scaling factor and  $\theta_0$  is the cart position defined by

$$\theta_0 = \text{Sgn}(\phi_c) \cdot G(r_c) \quad (10)$$

$$\phi_c = \tan^{-1}(f_2 \dot{x}) / (f_1 x) \quad (11)$$

$$r_c = \sqrt{(f_1 x)^2 + (f_2 \dot{x})^2} \quad (12)$$

As seen in (10)–(12), the dynamic behavior of the cart is described on the  $x$ - $\dot{x}$  phase plane. In the same manner as in the inference of pendulum angle,  $r_c$  and  $\phi_c$  in polar coordinates are used as the variables of the condition part in *if ~ then* rule.

## 2.3. Determination of Saturation and Gain Functions

The fuzzy logic is applied to determine the saturation functions and the gain functions. The basic inference rules for  $\text{Sat}(\phi_*)$  are

$$\begin{aligned} \text{If } \phi_* \text{ is } P_{\phi_*} \text{ then } \text{Sat is } P_s \\ \text{If } \phi_* \text{ is } N_{\phi_*} \text{ then } \text{Sat is } N_s \end{aligned} \quad (13)$$

where  $P$  and  $N$  denote *positive* and *negative*, respectively, and  $*$  denotes the subscripts  $p$  or  $c$ .

The basic inference rules for  $G(r_*)$  are

$$\begin{aligned} \text{If } r_* \text{ is } B_{r_*} \text{ then } G \text{ is } B_g \\ \text{If } r_* \text{ is } S_{r_*} \text{ then } G \text{ is } S_g \end{aligned} \quad (14)$$

where  $B$  and  $S$  denote *big* and *small*, respectively. Since the inputs  $(\phi_*, r_*)$  are directly measured, *singleton fuzzification* is used here. The outputs  $(\text{Sat}, G)$  are decided by using the *algebraic product-sum* and the *center of gravity* (COG) as fuzzification and defuzzification strategies. These input-output relations are shown in the form of membership functions in Fig. 2.1, in which  $(S_p, r_{pm}, u_m)$  and  $(S_c, r_{cm}, \theta_{0m})$  are design parameters for the FLC system, whose meanings will become clear in the later part.

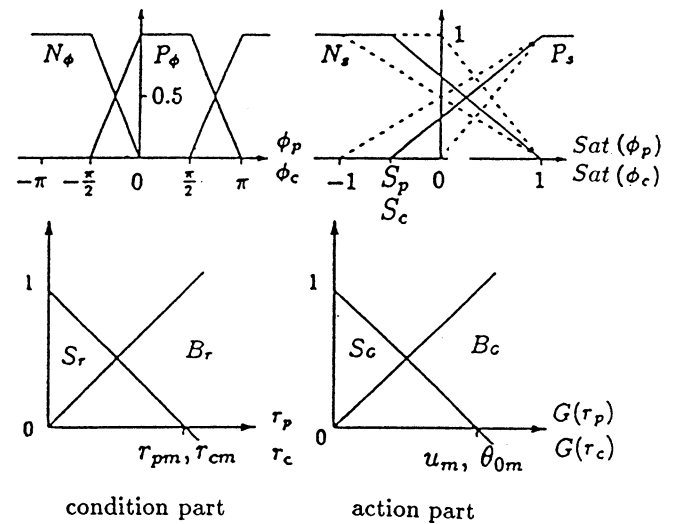


Fig. 2.1. Membership functions of rules (13) and (14)

The inference rules are used together with the control law given by (6), (9) and (10). The overall control system is shown in Fig. 2.2. As seen in this figure, the system can be interpreted as a cascade-type FLC system : the fuzzy logic controller in the outer loop shifts the set point for the pendulum control in the inner loop as a function of the cart position and velocity. Notice that as seen in Fig. 2.1, this FLC system can be interpreted as a variable structure system (VSS) <sup>2,8</sup>.

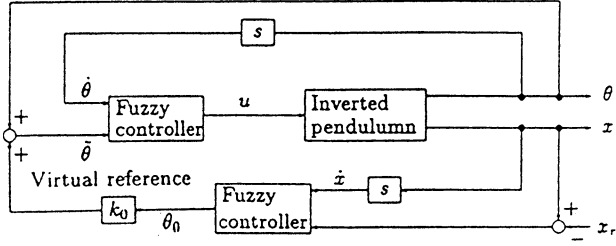


Fig. 2.2. Block diagram of FLC

### 3. Stability of Fuzzy Logic Control System for Inverted Pendulum

In this section, we analyze the local stability of the proposed FLC system for the pendulum. It is of interest to determine conditions of design parameters  $(S_p, r_{pm}, u_m)$ ,  $(S_c, r_{cm}, \theta_{0m})$  and scaling factors to assure the stability of the pendulum and cart control system.

#### 3.1. Stability of Pendulum

We first deal with the stability of the inverted pendulum with no constraint on the cart motion. Linearizing the motion equations (1) and (2) around  $\theta = 0$ , we obtain

$$(M + m)\ddot{x} + m l \ddot{\theta} + D_x \dot{x} = G_0 u \quad (15)$$

$$m l \ddot{x} + (m l^2 + J)\ddot{\theta} - m g l \theta + D_\theta \dot{\theta} = 0 \quad (16)$$

Ignoring the frictional coefficient  $D_x$  and eliminating  $\ddot{x}$ , we have

$$\ddot{\theta} = a_1 \theta + a_2 \dot{\theta} + a_3 u \quad (17)$$

where  $a_1 = m g l / a_0 > 0$ ,  $a_2 = -D_\theta / a_0 < 0$ ,  $a_3 = -m l G_0 / \{(M + m) a_0\} < 0$  and  $a_0 = J + m l^2 - m^2 l^2 / (M + m) > 0$ .

We apply the following transformation of variables to (17) :

$$\theta = r \cos \phi / k_1, \quad \dot{\theta} = r \sin \phi / k_2 \quad (k_1, k_2 > 0) \quad (18)$$

where the subscript  $p$  for  $\phi$  and  $r$  has been omitted for notational simplicity.

Then, we can describe the control law based on  $r$ ,  $\phi$  and  $S_p$  ( $-1 \leq S_p \leq 0$ ) as follows :

$$u = \text{Sat}(\phi, S_p) \cdot G(r) \equiv v \cdot r b \quad (19)$$

where  $\text{Sat}(S_p, \phi)$  and  $G(r)$  are rewritten by

$$\begin{aligned} \text{Sat}(S_p, \phi) &\equiv v \\ &= \begin{cases} 1 & (0 \leq \phi < \pi/2) \\ \left\{ -\frac{2(1-S_p)}{\pi} \phi + (2-S_p) \right\} & (\pi/2 \leq \phi < \pi) \\ -1 & (-\pi \leq \phi < -\pi/2) \\ \left\{ \frac{2(1-S_p)}{\pi} \phi + 1 \right\} & (-\pi/2 \leq \phi < 0) \end{cases} \quad (20) \end{aligned}$$

$$G(r) = r b \quad (21)$$

In (19), we define  $b = u_m / r_m$  with the relation  $r_m = k_1 \theta_{0m}$ , where  $u_m$  and  $r_m$  are defined in Fig. 2.1, and  $\theta_{0m}$  is the minimum value of  $\theta$  satisfying  $u = u_m$  when  $\dot{\theta} = 0$ . From (17) and (18), we have

$$\ddot{\theta} = \{\dot{r} - (1/k) r \cos \phi \sin \phi\} / (k_2 \sin \phi) \quad (22)$$

To analyze the stability of the closed loop system, we consider the following Lyapunov function candidate for the pendulum :

$$V \equiv V(\theta, \dot{\theta}) = (1/2) \cdot (\alpha \theta^2 + \dot{\theta}^2) \geq 0 \quad (\forall \alpha > 0) \quad (23)$$

Then, we have the time derivative of (23) along the solution of (17)

$$\begin{aligned} \dot{V} &= \dot{\theta}(\alpha \theta + \dot{\theta}) = (r/k_2)^2 \sin \phi \{ (a_1 + \alpha) k \cos \phi \\ &\quad + a_2 \sin \phi + a_3 k_2 b v \} \end{aligned} \quad (24)$$

Thus, the sufficient condition for the stable solution of (17) is obtained from  $\dot{V} < 0$ . As shown in **Appendix-A**, the inequality, we know that the inequality

$$b > -a_1 / (a_3 k_1) \quad (25)$$

is sufficient to make  $\dot{V}$  negative-definite in all the quadrants. It is critical to select the gain  $b$  so that the parameter  $u_m$  can be set large to satisfy (25).

### 3.2. Motion Analysis of Pendulum

We consider to analyze the pendulum motion in the proposed FLC system under the assumption of the stability condition (25). As shown in **Appendix-B**, the following two assumptions in the 2nd and 4th quadrants are made :

- [1] In the 2nd quadrant :  $-1 \leq S_p < a_1/(a_3 k_1 b)$  and there exist two solutions of  $h_2(\phi) = 0$ , and
- [2] In the 4th quadrant : there exist two solutions of  $h_4(\phi) = 0$ , where

$$h_2(\phi) = p_1 \cos^2 \phi + a_3 k_2 b \left\{ -\frac{2(1 - S_p)}{\pi} \phi + (2 - S_p) \right\} \cdot \cos \phi + (a_2/2) \sin 2\phi - 1/k \quad (26)$$

$$h_4(\phi) = p_1 \cos^2 \phi + a_3 k_2 b \left\{ \frac{2(1 - S_p)}{\pi} \phi + 1 \right\} \cos \phi + (a_2/2) \sin 2\phi - 1/k \quad (27)$$

where  $p_1 = a_1 k + (1/k)$ ,  $p_2 = a_1 k / (2p_1)$  and  $k = k_2/k_1$ . If  $-1 \leq S_p < a_1/(a_3 k_1 b)$  and  $h_2(\phi) = 0$  (or  $h_4(\phi) = 0$ ) has two solutions, then there exist one asymptote in the 2nd (or 4th) quadrant.

Under the assumption of (25), the pendulum trajectory is rotating clockwise in all the quadrants. In addition, if there exists the asymptote, the pendulum trajectory is going to the equilibrium point  $\theta = \dot{\theta} = 0$  along the asymptote.

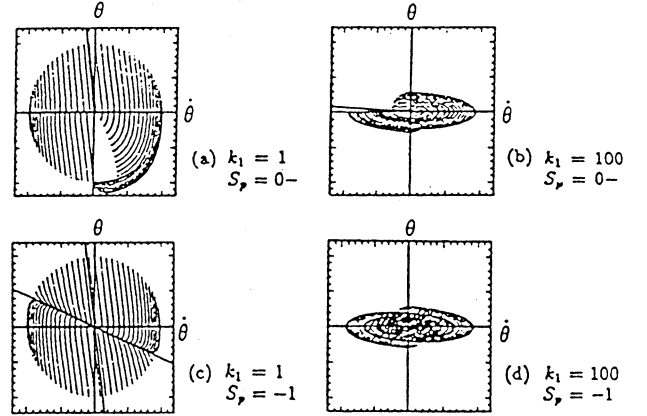
### 4. Simulations and Experiments

In this section, we validate the stability analysis for the proposed FLC system through some simulations and experiments with the design parameters and scaling factors listed in Table 4.1. The other design parameters are  $S_c = -1$  and  $r_{cm} = 0.4[m]$  and the physical parameters are omitted.

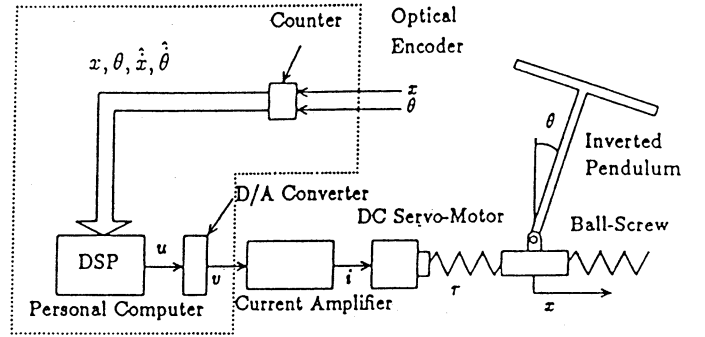
**Table 4.1.** Design parameters

	case 1 $k_1 = 1$	case 2 $k_1 = 100$
$k_0$	1	0.01
$k_2$	$\pi/6$	
$f_1$	30	
$f_2$	5.5	
$u_m$	10 [V]	
$\theta_{0m}$	$\pi/6$ [rad]	$\pi/600$ [rad]

### 4.1. Validation of Stability Analysis by Simulations



**Fig. 4.1.** Trajectories on phase plane



**Fig. 4.2.** Computer-controlled system

We check the stability of the pendulum control system. When the controller law is based only on the  $\theta$ - $\dot{\theta}$  phase plane, state trajectories are as shown in Fig. 4.1. The figure indicates results for each of cases 1 and 2 in Table 4.1 with  $S_p = 0-$  and  $S_p = -1$ .

[1]  $S_p = 0-$  : It is shown that the pendulum trajectory is constrained in the 2nd quadrant. This result justifies our analysis in 3.1 and 3.2 [See (a) and (b)].

[2]  $S_p = -1$  : Using the parameters of case 1, there exists an asymptote in the 2nd and 4th quadrants [See (c)]. On the other hand, using the parameters of case 2, the asymptote disappears [See (d)].

The control law for  $k_1 = 100$  and  $S_p = 0-$  gives a very similar action to human's one. This is, the cart is moving slowly toward the set point to decrease the pendulum velocity.

## 4.2. Experimental Results

We first consider an inverted pendulum, called *T-inverted pendulum* shown in Fig. 4.2<sup>4,9</sup>. We carried out the real-time control experiment of both the pendulum and the cart by using the proposed control law. Here, we used  $S_p = 0-$ ,  $S_c = -1$ ,  $r_{cm} = 0.4[\text{m}]$  and the values in case 2. In particular, a choice of  $S_p$  is important for controlling both the pendulum and the cart. As seen in Fig. 4.3, the values of  $\theta$  and  $\dot{\theta}$  are significantly small in this case than for  $S_p = -1$  when the trajectory is constrained along the asymptote in the 2nd quadrant. This corresponds to weakening the dynamic interaction between the pendulum and the cart. The behaviors of the pendulum and the cart are shown in Fig. 4.4. These results are a little different from those in the simulations based on the mathematical model described by (1) and (2), because of the presence of a dead band in the DC-motor used in the experiments. Figure 4.5 shows simulation results with this dead band. These resemble the experimental data in Fig. 4.4.

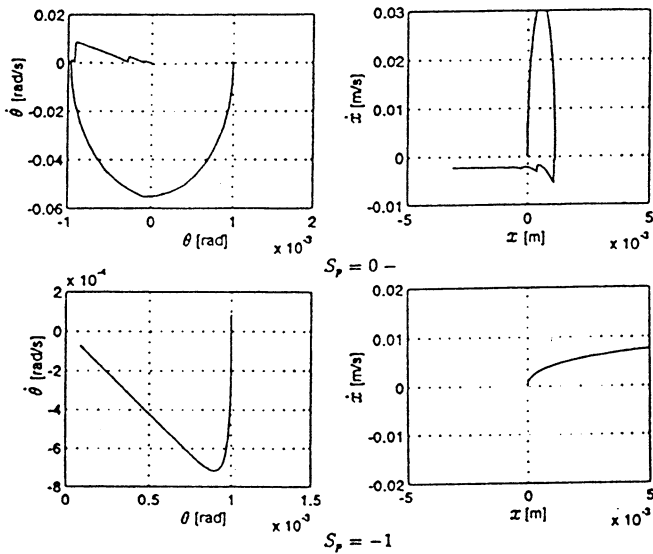


Fig. 4.3. Trajectories on phase plane

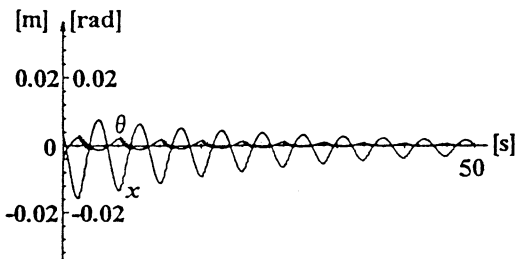


Fig. 4.5. Simulation results of control of T-inverted pendulum

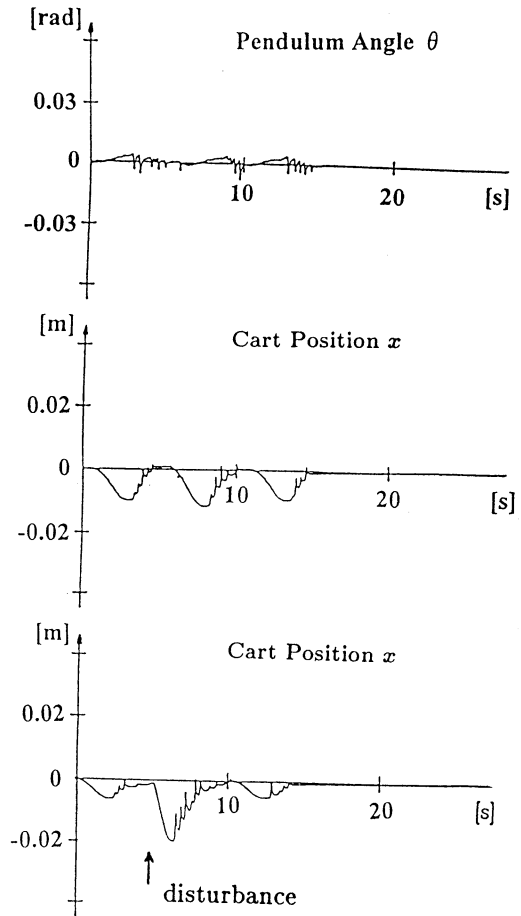


Fig. 4.4. Experimental results of real-time control of T-inverted pendulum

## 5. Conclusions

This paper presented a cascade-type FLC for an inverted pendulum whose stability was guaranteed by the Lyapunov method. By weakening the interaction between the pendulum and the cart by choosing the parameters  $k_0$ ,  $k_1$  and  $S_p$ , we realized a robustly stable control system without much information about the control objective.

## References

1. Yamakawa, T., "Stabilization of an inverted pendulum by a high-speed fuzzy logic controller hardware system," *Fuzzy Sets and Systems*, **32**, 161-180, 1989.
2. Kawaji, S., Shiotsuki, T. and Noguchi, I., "Fuzzy control system and its stability analysis for an inverted pendulum," *18th SICE Symp. on Control Theory*, 315-320, 1989 (in Japanese).
3. Ha, Y.-K. and Tomizuka, M., "Fuzzy global and local motion control of an inverted pendulum using multiple rule bases," *Proc. of the Asian Control Conf.*, 27-30, 1994.
4. Nakano, K., Eguchi, M. and Yamaguchi, Y., "Fuzzy reasoning control of inverted pendulum with considering of inter-



action with cart," *Trans. IEE of Japan*, **115-C**(1), 104-110, 1995 (in Japanese).

5. Passino, K.M. and Yurkovich, S., *Fuzzy control*, Addison-Wesley Longman Inc., 1998.

6. Palm, R., Drimiter, D. and Hellendoorn, H., *Model based fuzzy control- Fuzzy gain schedulers and sliding mode fuzzy controllers*, Springer-Verlag, Berlin, 1997.

7. Ostertag, E. and Corvalho-Ostertag, M.J., "Fuzzy control of an inverted pendulum with fuzzy compensation of friction forces," *Int.J.Systems Sci.*, **24**(10), 1915-1921, 1993.

8. Utkin, V.I., *Sliding modes in control and optimization*, Springer-Verlag, Berlin, 1993.

9. Nakano, K., et al, "Robust stabilization of an inverted pendulum system via rule-based sliding-mode generation," *13th IFAC World Congress*, F, 223-228, 1996.

## Appendix-A

We consider the condition for  $\dot{V} < 0$  when the pendulum state  $(\theta, \dot{\theta})$  is in the 1st and 3rd quadrants.

In the 1st quadrant, we set

$$\dot{V} = (r/k_2)^2 \sin \phi \cdot f_1(\phi) \quad (28)$$

Then

$$f_1(\phi) = (a_1 + \alpha)k \cos \phi + a_2 \sin \phi + a_3 k_2 b < 0 \quad (29)$$

leads to  $\dot{V} < 0$ . We can see that  $df_1/d\phi < 0$ , i.e.,  $f_1$  is monotonically decreasing. Therefore, if

$$f_1(0) = (a_1 + \alpha)k + a_3 k_2 b < 0 \quad (30)$$

then (30), i.e.,  $\dot{V} < 0$  holds.

On the other hand, we have

$$\begin{aligned} \dot{\phi} &= h_1(\phi) \\ &= p_1 \cos^2 \phi + a_3 k_2 b \cos \phi - 1/k + (a_2/2) \sin 2\phi \\ &< p_1 \cos^2 \phi - a_1 k \cos \phi - 1/k + (a_2/2) \sin 2\phi \\ &= p_1 (\cos \phi - p_2)^2 - (p_1 p_2^2 + 1/k) + (a_2/2) \sin 2\phi \\ &< 0 \end{aligned} \quad (31)$$

Thus, we conclude that the solution rotates clockwise on the  $\theta$ - $\dot{\theta}$  phase plane. This implies that the pendulum trajectory in the 1st quadrant moves toward the 2nd quadrant.

In the same manner, we can show that the similar results hold in the 3rd quadrant.

We next consider the condition for  $\dot{V} < 0$  in the 2nd and 4th quadrants. In the 2nd quadrant, we set

$$\dot{V} = (r/k_2)^2 \sin \phi \cdot f_2(\phi) \quad (32)$$

$$\begin{aligned} f_2(\phi) &= (a_1 + \alpha)k \cos \phi + a_2 \sin \phi \\ &\quad + a_3 k_2 b \left\{ -\frac{2(1-S_p)}{\pi} \phi + (2-S_p) \right\} \end{aligned} \quad (33)$$

Then, we can see that  $f_2(\phi)$  is downwards convex. Therefore, if  $f_2(\pi) = -(a_1 + \alpha)k + a_3 k_2 b S_p < 0$ , i.e.,  $b < (a_1 + \alpha)/(a_3 k_1 S_p)$ , then  $\dot{V} < 0$  holds. This implies that  $V$  can be always constructed so that  $\dot{V} < 0$ .

In the same manner, we can show that the similar results hold in the 4th quadrant.

## Appendix-B

We consider the existence conditions for asymptotes in the 2nd and 4th quadrants.

In the 2nd quadrant, we set

$$\begin{aligned} \dot{\phi} &= h_2(\phi) = p_1 \cos^2 \phi + a_3 k_2 b \\ &\quad \cdot \left\{ -\frac{2(1-S_p)}{\pi} \phi + (2-S_p) \right\} \cos \phi \\ &\quad + (a_2/2) \sin 2\phi - 1/k \end{aligned} \quad (34)$$

Then, the sign of  $\dot{\phi}$  and the corresponding behavior of solution are classified as follows :

(B-1)  $h_2(\pi) > 0$ , i.e.,  $-1 \leq a_1/(a_3 k_1 b) < S_p < 0$ , and there exists one solution of  $h_2(\phi) = 0$ .

(B-2)  $h_2(\pi) < 0$ , i.e.,  $-1 \leq S_p < a_1/(a_3 k_1 b)$ , and there exist two solutions of  $h_2(\phi) = 0$ .

(B-3)  $h_2(\pi) < 0$ , i.e.,  $-1 \leq S_p < a_1/(a_3 k_1 b)$ , and there exist no solutions of  $h_2(\phi) = 0$ .

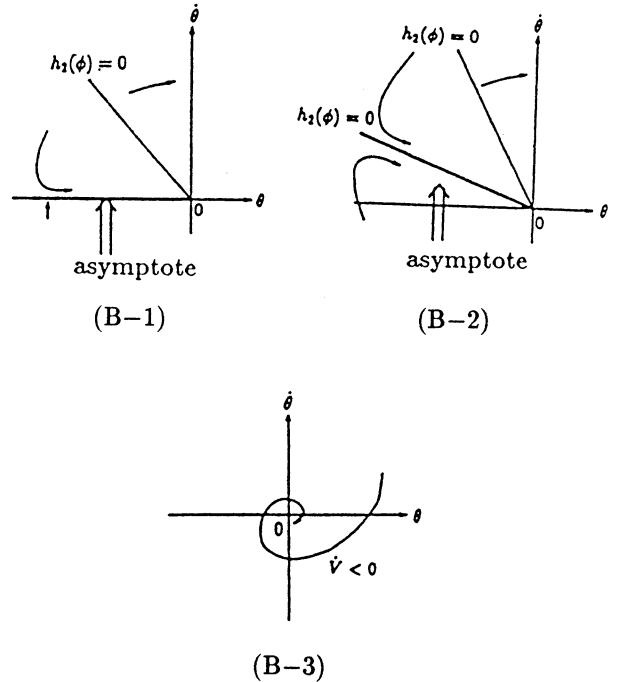


Fig. B-1. Behaviors of solution

The behaviors of solution in these three cases are illustrated in Fig.B-1. The asymptotes take place only under the condition that  $a_1/(a_3 k_1 b) < S_p < 0$  or  $-1 \leq S_p < a_1/(a_3 k_1 b)$  and there exists at least one solution of  $h_2(\phi) = 0$ . However, the asymptote on the boundary between the 2nd and the 3rd quadrant cannot be actually realized. Notice that  $h_2(\phi) = 0$  implies  $\phi = \text{const.} = \phi_0$ , where  $\phi_0$  is one of the asymptote angle.

The analysis in the 4th quadrant is performed in the same manner as in the 2nd quadrant.

# Application of neural and fuzzy control strategies for a mobile vehicle

Xin Wang, Masanori Sugisaka

*Department of Electrical and Electronic Engineering  
Oita University, 700 Dannoharu, Oita 870-11 Japan*

*Tel: +81-975-547831, Fax: +81-547841, E-mail: msugi@cc.oita-u.ac.jp, wangxin@cc.oita-u.ac.jp*

**Keywords:** mobile vehicle, control, neurocomputer, image processing

## Abstract

The goal of this paper is to submit a new idea from the practical point of view for control technologies. Our basic idea is to make a mobile vehicle (NMV) be capable of evaluating results of actions and modifying the inputs of the controller itself in order to drive optimal or near-optimal paths. In this paper, a control strategy, which is based on the combination of fuzzy and neural network control theories, is presented. The idea of the algorithm is to make the NMV learn driver's knowledge, and then drive from any a state to a desired state, and at the same time, running actions of the NMV are monitored and the inputs of system controllers are modified if necessary. The NMV system consists of a coupled-charged device (CCD) camera, a Von Neumann type computer and a neurocomputer, and image processing unit etc. The algorithm consists of a) making the driving models according to human knowledge and training neural networks or a neurocomputer, b) image recognition for the purpose of recognition of objects, c) tracking objects, d) driving actions of the NMV are monitored or modified by a fuzzy controller. This strategy made the NMV be capable of tracking a nominated object, avoiding obstacles in its way. The NMV can run special roads by the aid of three kinds of traffic marks. Both experimental results of the controller and only a fuzzy controller are given for the contrast.

## 1 Introduction

In recent years it has been recognized that to realize more flexible control systems it is necessary to incorporate other elements, such as logic, reasoning and heuristics into the more algorithmic techniques provided by conventional control theory, and such systems have come to be known as intelligent control systems. The general characteristics of intelligent systems is defined as having an ability to emulate human capabilities, such as planning, learning and adaptation[1]. Learning and adaptation especially are essential characteristics of intelligent control systems and, while adaptation

does not necessarily require a learning ability for systems to be able to cope with a wide variety of unexpected changes and environments, learning is invariably required. The field of intelligent control is multidisciplinary and combines techniques from control theory, computer science, psychology, physiology and operations research. The research of navigation of vehicles has lasted for many years. J. P. Laumond et al[2] applied three-steps algorithm to a mobile robot and solved the problem that how to compute optimal feasible paths in a configuration space (the space in which the motion of rigid bodies amid obstacles in the 3-D dimensional Euclidean space is translated into the motion of point in some space) equipped with the singular metric induced by the nonholonomic constraints. Y. M. Enab[3] build a controller for the unknown nonlinear time invariant dynamic process based on modeling the behavior of its human operator. D. Gorinevsky et al[4] designed a controller for an automated support system and performed desired features of such a system and parking controller. The navigation problems can be classified into parking problems, motion planners, vehicle sensors, image processing methods and so on. We pay attention to two points, strategies of driving feasible paths and a practical image processing method by a CCD camera sensor.

Though the Maximum Principle in the framework of Optimal Control Theory can compute time optimal paths with constraint conditions, it is not easy for an arbitrary nonholonomic system to compute a time optimal path by using this principle and only approximate solutions are given in general cases. The goal of this paper is to present a new control strategy for the navigation of the NMV from the practical point of view. Our basic idea is to use the human knowledge to make the NMV be capable of learning, evaluating results of actions and modifying the inputs of the controller itself in order to acquire optimal or near-optimal results.

In this paper, the obstacles were assumed not more than one in an image processing sampling period. Some other practical issues considered include the boundedness of the steering angle, and optimization of control inputs in the sense of minimizing the steering effort required to drive this NMV. We describe the control process and intelligent algorithm and present the experiments and discussion, then we present our conclusions.

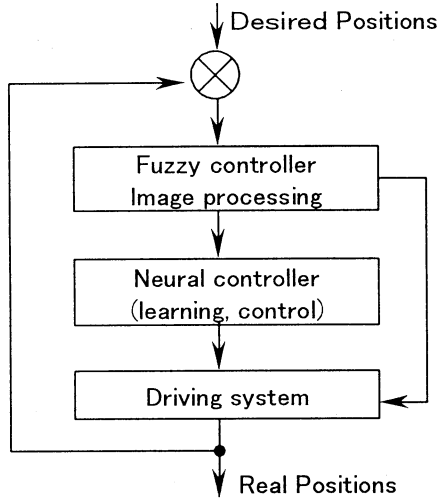


Fig. 1: Block diagram for the control system

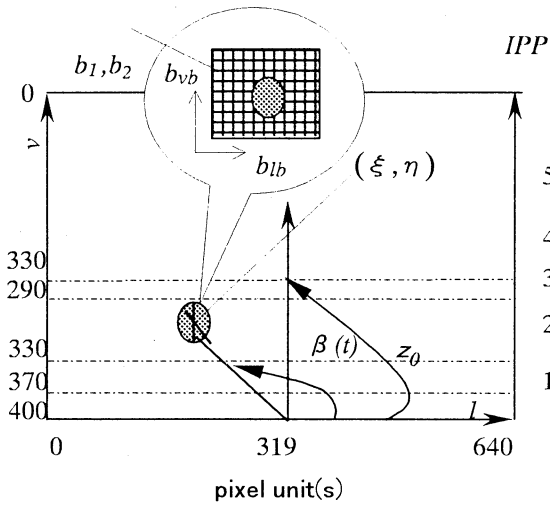


Fig. 2: The image processing method

## 2 Control Process of the System

The block diagram of the NMV control system is shown in Fig. 1. The idea of the intelligent algorithm is to make the NMV learn driver's knowledge, and then drive itself from any a state  $q(t)$  to a desired state  $q^d(t)$ , detect and modify the inputs of the controllers if necessary at the same time. The fuzzy system or fuzzy controller is to imitate the human reasoning process to ill-defined systems, and is able to realize the indefinite reasoning process without using exact quantitative analysis. The NMV system consists of three main parts. (1) neurocomputer used for driving and recognition, (2) fuzzy controller, (3) image processing units. The intelligent algorithm consists of making the driving models according to human knowledge and training the NMV or neurocomputer, driving control using a neurocomputer, image processing for the purpose of recognition

of objects, and driving actions monitored and modified by a fuzzy controller.

**Neurocomputer RN-2000** The neurocomputer RICOH-2000 which is based on the back propagation algorithm is used in the control system. The back propagation network(BPN) is capable of changing the inputs-outputs problem of a set of sample into a nonlinear optimal map[5]. The RICOH-2000 consists of seven RN-200 digital neural network VLSI chips and sixteen neurons in each layer, and whole 256 synapses are integrated in a  $13.73 \times 13.73 \text{ mm}^2$  VLSI chip. This chip can perform 5.12 giga pulse operations per second. It corresponds to effective neural computing rate of 40M CPS[6][7]. Let  $\Lambda = \{\lambda_1(i), \dots, \lambda_{m_1}(i)\} \in R^{m_1}$  be a vector containing  $m_1$  measurements representing the process states at time index  $i$ ,  $\Gamma = \{\gamma_1(i), \dots, \gamma_{n_1}(i)\} \in R^{n_1}$  be the outputs which response the inputs. The RICOH-2000 realized the map  $\mathcal{R}: \Lambda \rightarrow \Gamma$ . 12 driving patterns and three patterns of marks were used and  $m_1 = 8$  and  $n_1 = 1$  in our controller.

**Control process** The intelligent strategy includes two parts, tracking a nominated object without obstacles and avoiding an obstacle when tracking a nominated object. All of the information about the environments is caught by a CCD camera. The information is affected by distinguishability, image interval of pixels and, brightness of the environments and so on. Perfect motion planers are not easy for an arbitrary nonholonomic system. Therefore, the practical intelligent method is valuable. There are  $600 \times 400$  pixels used in an image processing sampling period in our design. An image unit after the image processing which inputs are from the CCD camera of the NMV is divided into 256 tones.

Firstly, in order to catch a nominated object and an obstacle, five image processing parts (IPPs)(shown in Fig. 2) including 43 image blocks (IBs) which width and height are  $b_l$  and  $b_v$  (shown in Table 1) are established on the base of regular image patterns between real distance and pixel distance[8]. The scan image unit is from the IBs in the minimum IPP to the ones in the maximum IPP one by one. In Table 1  $b_1$  and  $b_2$  are the coordinate of image areas in  $v$  direction,  $\Delta b_w = b_2 - b_1$ ,  $b_l$ ,  $b_v$ ,  $b_{lb}$ ,  $b_{vb}$  are the pixel units in  $l$  and  $v$  direction shown in Fig. 2 respectively, and  $\Delta x_{lb}$ ,  $\Delta x_l$  are pixel intervals in  $l$  direction,  $\Delta x_{vb}$ ,  $\Delta x_v$  are pixel intervals in  $v$  direction. Symbols with same notes response each other in Table 1. For example, if we want to calculate one of a block pixel distance (PIXD) in  $l$  direction when  $\text{IPP}=1$ , then  $\text{PIXD} = b_{lb} \times \Delta x_{lb} = 16 \times 4 = 64$  pixels. The obstacles are recognized according to their colors and shapes. The colors are distinguished by means of graded-stage way. If the color of an object, which is after image processing, is caught, the center of gravity  $(\xi, \eta)$  of the object will be calculated and an expanded block which center is  $(\xi, \eta)$  will be formed (shown in

Table 1: Parameters of Image Processing

IPP	$b_1$	$b_2$	$\Delta b_w$	$b_{lb}$	$b_{vb}$	$b_l$	$b_v$	$\Delta x_{lb}$	$\Delta x_{vb}$	$\Delta x_l$	$\Delta x_v$
1	370	400	30	16	16	128	30	4	3	1	1
2	330	370	40	16	16	80	40	3	2	1	1
3	290	330	60	16	16	64	60	2	2	1	1
4	270	290	20	16	16	32	20	1	1	1	1
5	0	270	270	16	16	128	270	3	2	1	1

Fig. 2). Therefore, compressed image for recognition using moment invariant computation are finished.

The moment invariant computation has been used in image recognition successfully[9]. We use the results in this paper. Details are described in the reference[9]. The  $16 \times 16$  two-value compressed image are used to calculate the moment invariant  $Q_i$  ( $i=1, \dots, n$ ). In our case,  $n = 7$ . After being standardized, the moment invariant  $Q_1 \sim Q_7$  became the inputs of the *RN-2000* neurocomputer.

Variables of the fuzzy controller are decided as follows[10]. Let the angle of  $(\xi, \eta)$  be  $\beta(t)$  ( $\geq 0$ ), let  $\beta(t) = \beta_{\min\{IPP\}}(t)$  ( $\min\{IPP\}$  means a minimum IPP if an object crosses more than one IPP). The  $i_{th}$  fuzzy logic rule  $Rule_i$  is described as follows[11].

$Rule_i$ : if  $z_1(t)$  is  $A_1^i$  and  $z_2(t)$  is  $A_2^i$  then  $\phi(t)$  is  $B^i$   
 $(i = 1, 2, 3, \dots, n)$

where  $z_1(t)$  and  $z_2(t)$  are inputs and  $\phi(t)$  is the output of the fuzzy controller at the  $t$  time index.  $A_1^i, A_2^i, B^i$  are fuzzy sets in the  $i_{th}$  fuzzy rule[12]. In the NMV system, the output  $\phi(t)$  is the steering angle. One of two inputs  $z_1(t)$  is the angle between  $\beta(t)$  the angle of the center of gravity of an object  $(\xi, \eta)$  and a reference position  $z_0$ .  $z_1(t) = \beta(t) - z_0$ , where  $z_0 = 90^\circ$ . Another input is the rate of  $z_1(t)$ .  $z_2(t) = \Delta z_1(t) = z_1(t) - z_1(t-1) = \beta(t) - \beta(t-1)$ . The role of the fuzzy controller is to monitor continuous changes of the center of gravity of an object after image processing and make the proper modifications.

### 3 Experiments and Discussion

The NMV was installed with a microcomputer NEC PC9821Ap/U7 (NEC, Tokyo, Japan), some I/O boards including a video interface SUPER CVI (Canopus Co Ltd., Tokyo, Japan). The NMV's two rear wheels can rotate forwards and backwards about the axle shaft and are driven by a 24V DC motor using cross helical gears (Denken Engineering Co Ltd., Oita, Japan). The NMV's front wheels are driven by a 24V DC stepping motor (Oriental Motor Co Ltd., Tokyo, Japan) for motion to the left or right, with transmission being via gear wheels. Two 6V DC motors (Oriental Motor Co Ltd., Tokyo, Japan) are employed to rotate the CCD

camera in the horizontal and vertical directions. The location of the front wheels and camera are detected by three encoders (Omron Corporation, Tokyo, Japan).

Figure 3 shows our experiments by a fuzzy strategy and the intelligent strategy respectively. The up one in Fig. 3 shows the results using a fuzzy controller, the initialized statements are  $[(x_1(t), x_2(t), \theta(t)) = (1.0, 1.0, 30^\circ)]$ , where  $x_1(t)$  and  $x_2(t)$  are shown in Fig. 3,  $\theta(t)$  is the angle between the center line of the NMV and  $x_1(t)$  at time index  $t$ , and the aimed position is  $(12.0, 2.0, 60^\circ)$ . The down one in Fig. 3 shows the results using the intelligent controller. Figure 3 also give a trajectory including an obstacle (illuminated by a dark area), the initialized statement is  $(1.0, 1.0, 30^\circ)$ , the terminations are in  $(12.0, 2.0, 60^\circ)$  and  $(12.0, 2.5, 60^\circ)$  respectively. The desired ways in the experiments of the fuzzy control and intelligent control are the same. On the curved part, the intelligent method proved obvious advantages. This is not to understand because it is necessary for a driving procedure to be adjusted continuously in order to adapt changing conditions. The intelligent strategy not only modified results of actions but also denied unreasonable judges and avoid doing unreasonable things as possible as it could. The initialized statements can represented the common cases of tracking problems. The driving sampling time is 20ms depended on the structure of the NMV. The NMV adjusted the control inputs to decrease the position errors, and runs near optimal paths. The maximum speed of the NMV is about 2.0m/s. The errors resulted from (1) the data of image information from CCD camera are not precise, (2) regular image patterns between real distance and pixel distance are approximate and, (3) there are errors between teaching patterns and learning results, (4) the mechanical and electrical inertia and so forth.

### 4 Conclusions

An strategy which is based on the combination of fuzzy and neural network control theories was applied into the navigation of a NMV successfully. Technologies of both neurocomputer and Neumann type computer were utilized in the control system, and made the NMV be capable of learning, remembering, and be able to drive and check the effect of actions according to

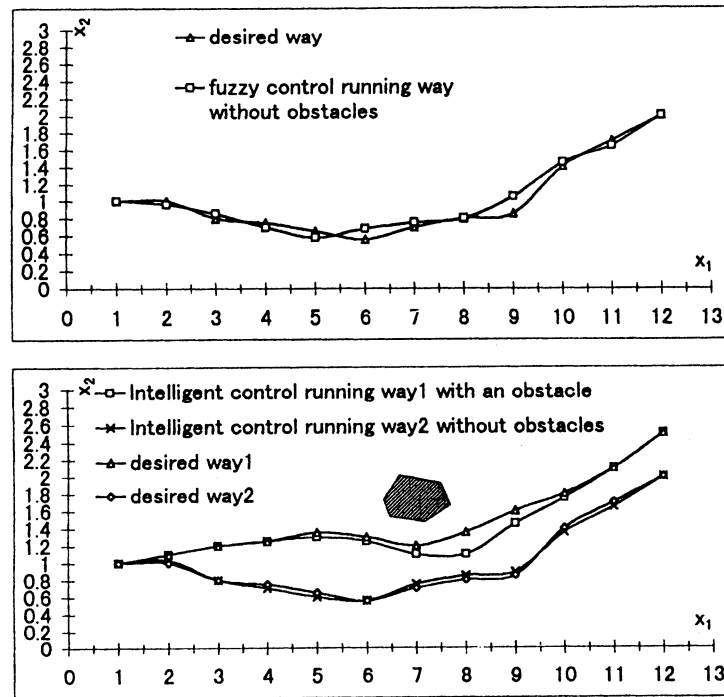


Fig. 3: Experimental results

some rules and make things better. The image processing strategy can also adapt to other cases for example navigating transport in plants, parking problem by the aid of landmarks, a curve way in farms etc [5]. Our research will focus on applying the developed technique into a electrical car which maximum speed can reach to 60 km/hour.

## Reference

- [1] Linkens, D. A., Nyongesa, H. O.(1996) "Learning System in Intelligent Control: An Appraisal of Fuzzy, Neural and Genetic Algorithm Control Application", *IEE Proc. Control Theory Appl.*, 143(4), pp.367-386
- [2] Laumand, J. P., Jacobs, P. E.(1994) "A Motion Planner for Nonholomic Mobile Robots", *IEEE Trans. on Robotics and Automation*, 10(5), pp.577-593
- [3] Enab, Y. M.(1996) "Intelligent Controller Design for the Ship Steering Problem", *IEE Proc. Control Theory Appl.* 143(1), pp.17-24.
- [4] Gorinevsky, D., Kapitanovsky, A., and Goldenberg, A.(1996) "Neural Network Architecture for Trajectory Generation and Control of Automated Car Parking", *IEEE Trans. on Control System Tech.* 4(1), pp.50-56
- [5] Jiao, L. C. *System Theory of the Neural Network*, Xian University of Dianzikeji Press, Xian(1995).
- [6] Sugisaka, M.(1997) "Development of a Mobile Vehicle", *Science of Machine*(in Japanese), 49(6), pp.8-15
- [7] Sugisaka, M. et al.(1994), "Hardware Based Neural Identification: Linear Dynamical Systems", *Proc. of International workshop on intelligent systems and innovative computations*, pp.124-131
- [8] Sugisaka, M., Wang, X.(1997) "Control Strategy for Smooth Running a Mobile Vehicle with Neurocomputer", *Twelfth International Conference on Systems Engineering*, 2(2), pp.664-669
- [9] Sugisaka, M., Teshnehlal, M.(1993) "Fast Pattern Recognition by Using Moment Invariants Computation via Artificial Neural Networks", *Control-Theory and Advanced Technology*, 9(4), pp.877-885
- [10] Sugisaka, M., Wang, X.(1996) "The Development of a Practical Control Method for an Intelligent Mobile Vehicle", *Proc. of Int. Symposium on Artificial Life and Robotics*, pp.234-237
- [11] Sugeno, M. *Fuzzy Control*, Nikkankougyo Sinbun-shya(in Japanese), Tokyo(1989)
- [12] Madan, M. Rommohan, K., Ronald R., *Advances in Fuzzy Set Theory and Applications*, North-Holland, New York(1990)

# The Time Minimum Optimization Strategy of Idiotype Immune Type Signal Transmission Network System

H Hirayama, and \*Y. Okita

Department of Public Health Asahikawa Medical College.  
4-5 Nishi-Kagura, Asahikawa city 078 Japan.

Tel 0166-65-2111 ex 2411. Fax 0166-68-2419.

\* Department of Education Technology. National Institute of Special Education. Nobi-5-1-1. Yokosuka

## Abstract

We have hypothesized the time minimum optimal control strategy for an idiotype immune type signal transmission system so as to apply the shortest time optimization principle to the network information processing system that have immune properties. The idiotype immune reaction system was described by sequential non linear differential equations. Stimulation and secretion processes of an antibody in the idiotype network were characterized by the magnitude of cross-linking of the complemental antibody and B cell receptor. The changes in lymphocytes, antibody and memory cell can explain the immune system must react as soon as possible so as to eliminate the exogenous disturbance. The present investigation will be available for producing the information processing system having immune properties.

Idiotype. Immune system, Time minimum control. Optimal control, Information processing.

## 1. Introduction.

An entire scope of the network behavior [1] of the immune reaction can never be disclosed by biological experiments because of the complicated interaction among the species of the network. This complexity can only be analyzed through a computational approach. The present computation is intended to disclose the transient behavior of an idiotype network [1] of immune system comprised of bone marrow origin lymphocyte (B cell ).

## 2. Mathematical Modeling.

### 2-1. Stimulation and secretion of antibody in idiotype.

We assume that the B cells constitute an idiotype network composed of  $n$  sub populations. The B cells in the  $i$  sub population  $B_i$  are assumed to grow by [2]

$$\frac{\partial B_i}{\partial t} = m_i + p_i F_i(h_i) B_i - dB_i \quad (1)$$

where

$B_i$  : the number of the B cell in the  $i$  sub population.

$m_i$  : the producing rate of the  $B_i$  cell in bone marrow

$dB_i$  : the turn over rate of the  $B_i$  cell (/day).

$h_i$  : the magnitude of cross-linking between an antibody and a B cell receptor

$p_i$  : the proliferating rate (/day) of the  $B_i$  cell being stimulated by an antigen or an antibody.

$F_i$  : the fraction of the  $B_i$  cell being stimulated to grow.  $F_i$  describes the activating process of the B cells and is a function of  $h_i$ .

### 2-2. Affinity function [2].

The magnitude of cross linking connectivity between the antibody and the B cell receptor,  $h_i$  is a function of molecular affinity  $J_{ij}$  between an antibody  $A_j$  and a  $B_i$  cell provided their structures are complementary. Hence,  $J_{ij}$  determines  $F_i$  through  $h_i$ . Any B cell sub population ( $B_i$  for example) is stimulated by being bound through their receptors with all the possible complementary anti-idiotypic antibodies in the network. This is because any idiotype is recognized as an "antigen" by all other idiotype antibodies and B cells. Thus, the connectivity  $h_i$  represents an integrated effect of all the related anti-idiotypic antibodies with various weighted affinities such that

$$h_i = \sum_{j=1, j \neq i}^n J_{ij} A_j \quad (2)$$

where  $J_{ij}$  is an weighted affinity matrix. The weight is slightly different from an antibody to antibody. This difference owes to the subtle structural difference among the Fab portions of the antibodies. This difference derived from minor fluctuation or chaotic change in the re-combination of immune globulin genes [3]. We assume that the affinity of an antibody  $A_j$  to other antibody  $A_i$  is the same as its affinity to the receptor on  $B_i$  cell,  $J_{ij} = J_{ji}$ . We set all the elements of  $J_{ij}$  unity and the degree of this matching simply by high and low levels.

2-3. Activating and secretion processes based on threshold.

We use an exponential form for the activating function  $F_i(h_i)$  to describe increasing and saturating phases of the immune reaction [4].

$$F_i(h_i) = \frac{1}{1 + e^{-(h_i - h_0)}} \quad (3)$$

where  $h_0$  is the threshold for activating the immune reaction. The present activating function does not describe the high concentration inhibition effect nor tolerance effect [5]. In the course of immune response, stimulation of a representative of class "i+1" occurs via a single specie on level "i" whereas suppression for the species "i" is promoted by a whole class "i+1" due to the idiotypic network. Therefore, the threshold for activation should be set higher and be more complicated than the threshold for suppression. Thus, the shape of the activating function can not be a symmetric bell shape as Boer proposed [6]. Another reason to discard the high concentration inhibitory effect relates to the time span to induce the high concentration tolerance of the immune reaction [7]. Successive high doses of an antigen are required to convert the mode of transient immune response from the low concentration tolerance zone through the normal

response to the high zone tolerance. For inducing the high concentration tolerance of immune reaction, therefore, antigen has to be supplied so that its concentration has been kept constant over a period of a number of weeks. Such successive stimulation by an antigen, however, alters the property of the idiotypic network during the long term stimulation. The present computational analysis was confined to hours time span so as to apply the present computational approach to the routine clinical laboratory examination for the immune system. Thus, we did not take into consideration of the high concentration tolerance effect and used the present asymmetric form of  $F_i$ .

The stimulated  $B_i$  cells release antibody,  $A_i$ . The rate of release of free antibody,  $s_i(t)$  [5] is

$$\frac{\partial s_i(t)}{\partial t} = k_{si} (s_{\max i} F_i(h_i) - s_i(t)) \quad (4)$$

where

$k_{si}$  : the rate of inducing antibody  $A_i$  (nano mole / cell / day).

$s_{\max i}$  : the maximum rate of releasing antibody  $A_i$  (n mole / cell / day).

The concentration  $S_j$  of free receptor site on the  $B_j$  cell can be expressed in terms of the total number  $R_t$  of receptor site on the  $B_j$  cell membrane.

Fig 1-a

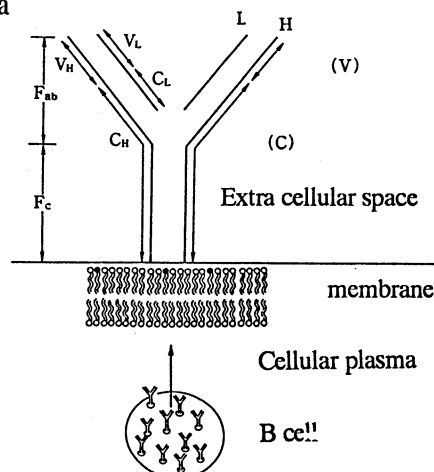


Fig 1-b

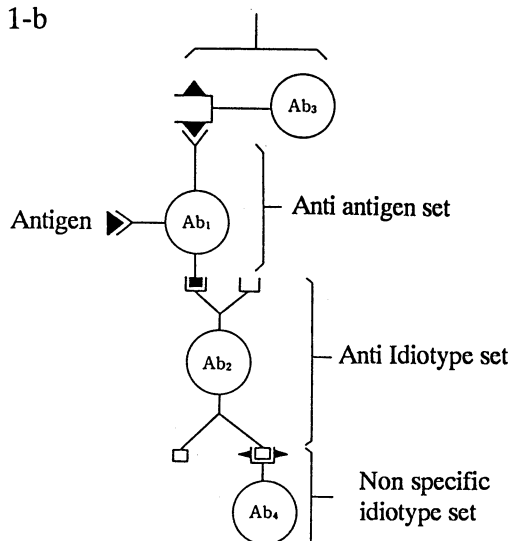
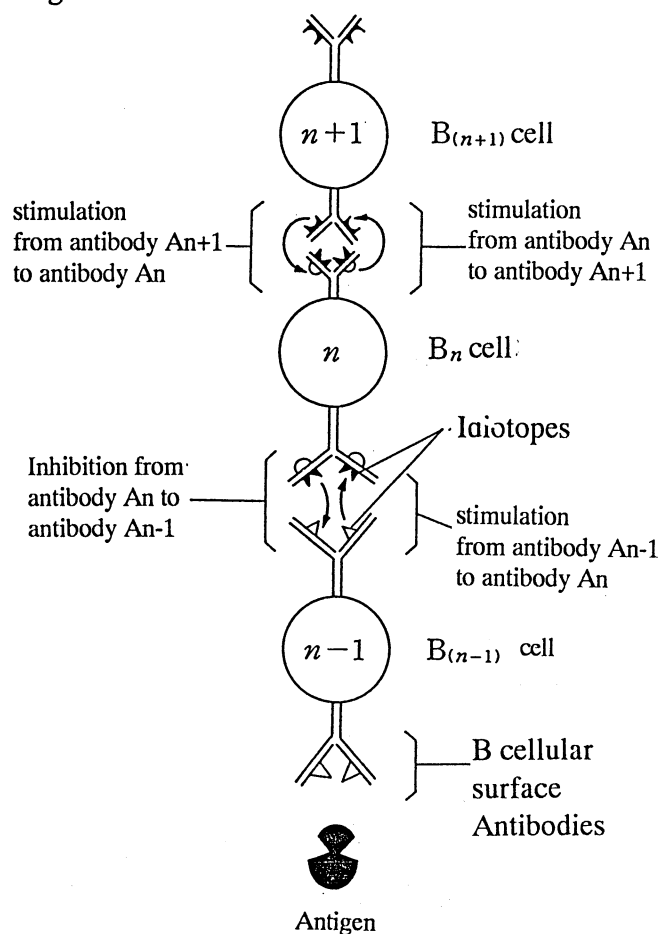


Fig 1. A schema of Idiotypic antibody system

Fig 1-c



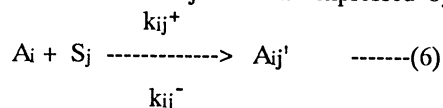
$$S_j = B_j R_t - \sum_{i=1}^n A_{ij}' - 2 \sum_{i=1}^n A_{ij}'' \quad \text{-----}(5)$$

$A_{ij}'$  is the concentration of the antibody  $A_i$  either of its two Fab arms linked to one receptor site on the  $B_j$  cell membrane. We call  $A_{ij}'$  a "single bound antibody".  $A_{ij}'$  has still another free Fab arm.  $A_{ij}''$  is the concentration of the antibody  $A_i$ , both of its two Fab arms have linked to the receptor sites on the  $B_j$  cell. We call  $A_{ij}''$  a "double bound antibody". The product of  $B_j$  and  $R_t$  equal to the total receptor sites on the  $B_j$  cell.

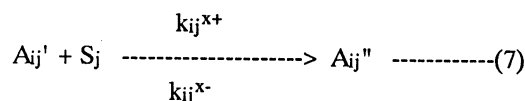
## 2-3. The temporal behaviors of idiotypic immune network.

We describe a general case of interacting two B cell sub populations,  $B_i$  and  $B_j$ . B cells in the  $i$  sub population recognize antigen,  $G$ . B cells in the  $j$  sub population have receptors which molecular structures are complementary to the receptors on the  $B_i$  cells. Therefore, the  $B_j$  cell receptor is complementary to the Fab part of the antibody  $A_i$ . Antigens bind to the receptors on the  $B_i$  cells, cross link and stimulate them to produce the antibody  $A_i$ . The antibody  $A_i$  binds to the complementary receptor on the  $B_j$  cell membrane.  $A_i$  cross links the  $B_j$  cells and lead them to produce the antibody  $A_j$ .  $A_j$  is a kind of an anti idiotypic

antibody against the antibody  $A_i$  and the  $B_i$  cell. In turn, the Fab part of the  $A_j$  can bind to the receptor on the  $B_i$  cell membrane. Further, the Fab part of  $A_j$  can bind with the Fab part of  $A_i$  to form an immune complex  $C$ . The binding process of either of the two free Fab arms of the antibody  $A_i$  to the receptor site on the  $B_j$  cell is expressed by



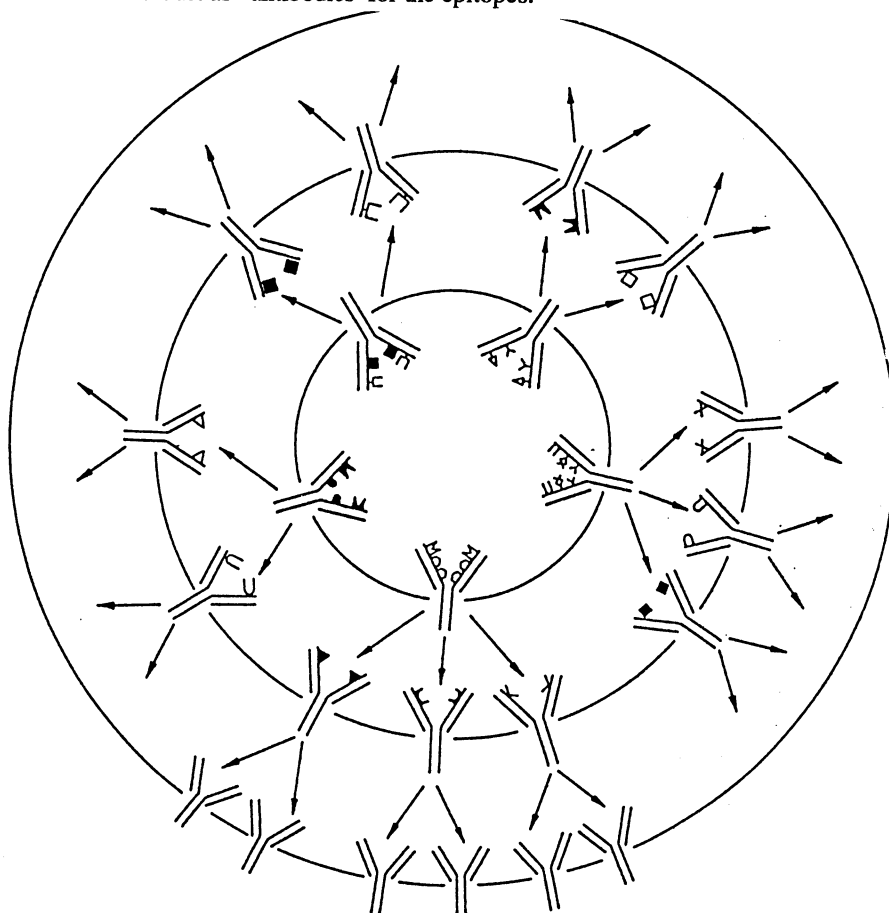
where  $k_{ij}^+$  is the rate constant for binding one of two Fab arms of  $A_i$  to one receptor site on the  $B_j$  cell. The residual free Fab arm of  $A_{ij}'$  can still bind to other receptor site on the  $B_j$  cell. Hence, a receptor cross linked complex (double bound antibody)  $A_{ij}''$  on the  $B_j$  cell is



where  $k_{ij}^{x+}$  is the rate constant for binding the residual free Fab arm of  $A_{ij}'$  to one receptor site on the  $B_j$  cell.

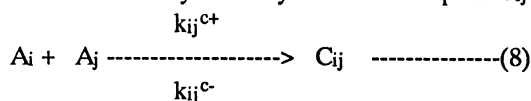
**Fig 2. An Idiotypic Ring of immune network system.**

This figure shows the spread of idiotypic immune reactions among the idiotypic B cells or antibodies. The antibodies on the circles having the same radius are the same generation characterized by the different epitopes. The antibody ring that encircle the ring has corresponding molecular structures that act as antibodies for the epitopes.





Antibody  $A_j$  behaves as an antigen against the  $i$  sub population of the  $B_i$  cells and stimulates them. Reaction between the antibodies  $A_i$  and  $A_j$  which form an antibody-antibody immune complex  $C_{ij}$  is



where  $k_{ij}^+$  and  $k_{ij}^-$  are the rate constants for binding and dissociating reactions.

The temporal change in the concentration of the antibody  $A_i$  is

$$\begin{aligned} \frac{\partial A_i}{\partial t} = & s_i B_i - d A_i A_i - 2 \sum_{j=1}^n k_{ij}^+ A_i S_j + \sum_{j=1}^n k_{ij}^- A_{ij}' \\ & - 4 \sum_{j=1}^n k_{ij}^+ A_i A_j + \sum_{j=1}^n k_{ij}^- C_{ij} \quad (9) \end{aligned}$$

The temporal change in a single bound antibody,  $A_{ij}'$  is

$$\frac{\partial A_{ij}'}{\partial t} = 2 k_{ij}^+ A_i S_j - k_{ij}^- A_{ij}' - k_{ij}^+ A_{ij}' S_j / B_j + 2 k_{ij}^- A_{ij}'' \quad (10)$$

About  $A_{ij}''$  and  $C_{ij}$

$$\frac{\partial A_{ij}''}{\partial t} = k_{ij}^+ A_{ij}' S_j / B_j - 2 k_{ij}^- A_{ij}'' \quad (11)$$

$$\frac{\partial C_{ij}}{\partial t} = 4 k_{ij}^+ A_i A_j - k_{ij}^- C_{ij} - d_{cij} C_{ij} \quad (12)$$

We set the shortest time reaction of the immune system as a control strategy for the defensive reaction. The cost function of this time minimum optimal control is

$$J(u) = \int_0^T 1.0 \, dt = T \quad (13)$$

The optimized differential equations for the co-state variable  $\lambda_n$  that corresponds to the state equations can be obtained by differentiating the Hamiltonian with respect to the corresponding state variables.

$$\frac{\partial \lambda_n}{\partial t} = - \frac{\partial H}{\partial X_n} \quad (14)$$

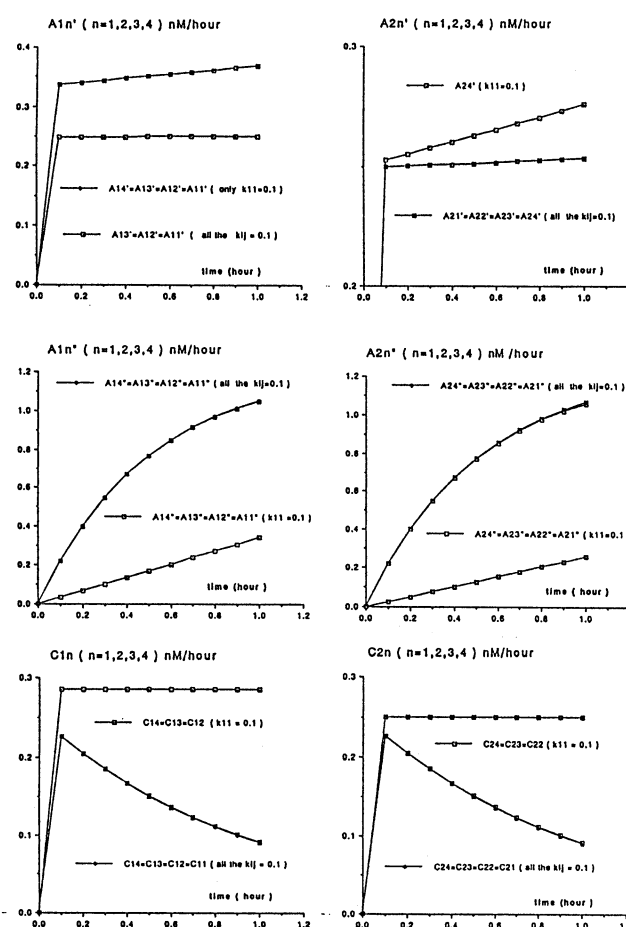
where  $X_n$  are the state variables assigned for the concentration of the species in the idio type network. We solved the optimized state equations and co-state equations by the numerical method in which rate constants are derived from the reported experimental data. [6],[8].

### 3. Results and Discussion.

Fig 3-a shows the temporal changes of concentration (nM/hour) of idio type single bound antibodies,  $A_{in}'$  ( $i=1,2, n=1,2,3,4$ ). Fig 3-b shows the temporal changes of concentration (nM) of double bound antibodies,  $A_{in}''$  ( $i=1,2, n=1,2,3,4$ ) and Fig 3-c gives those of immune complex  $C_{in}$  ( $i=1,2, n=1,2,3,4$ ). With the progress of the reactions, the double bound antibodies as a completed form of the idio type immune reaction increased linearly.

In the present work, we have confined analysis only for a short time span to evaluate the present computational approach by comparing to experiments in the future. Because comparison to the experimental data in a long time span makes the problem more complicated. By accessing to the quantitative experimental data the present

Fig 3



computational approach will be verified its usefulness in creating information transmission network system having auto defense mechanisms against the exogenous disturbance and mutual organization property.

### 4. Conclusion.

We proposed an immune type network system characterized by the idio type structure for creating new information transmission system having auto defense mechanism and mutual interactive organization property.

### 5. References.

- [1] Jerne, N.K. (1974) Toward a net work theory of the immune system. *Ann. Immunol* 125, 373-389.
- [2] Boer, R. J., Keverkiis, I. G. and Perelson, A. S. (1990) *Chemical Engineering Science* 45, 2375- 2382.
- [3] Tonegawa, S.(1983) Somatic generation of antibody diversity. *Nature* 302, 575-581.
- [4] Perelson, A. S. (1989) Immune network theory. *Immunological. Reviews* 110, 5- 36.
- [5] De Boer, R.J. (1991) Size and connectivity as emergent properties of a developing immune network. *J. Theoretical. Biol* 149, 381-424.
- [6] Boer, R. (1993) Immune network behavior. *Bulletin. Mathematical Biology* 55, 745-780.
- [7]. Richter, P.H. (1975) A network theory of the immune system. *Eur. J. Immunology* 5, 350-354.
- [8] Delishi, C. and Metzger, H. (1976) Some physical chemical aspects of receptor-antibody interactions. *Immunological. Communications* 5, 417-446.

## An Application of Multi-modal Neural Network to Multiple Control System

T. Nakagawa, \*M. Inaba, K. Sugawara, I. Yoshihara and K. Abe

Graduate School of Engineering, Tohoku Univ. Sendai 980-8579, Japan

nakagawa@largesys.ecei.tohoku.ac.jp

\*Systems Development Laboratory, Hitachi,Ltd Kawasaki,215-0013, Japan

\*inaba@sdl.hitachi.co.jp

### Abstract

We proposed Multi-modal Neural Network(MNN) in order to enhance nonlinear characteristics of multilayer neural networks[1]. Our aim is to improve control performance of multiple control system[2] whose candidate controllers consist of multilayer neural networks.

We applied MNN to making controllers for cart-pole balancing system[3]. We generate candidate controllers which are designed for a certain environmental condition. The results are that MNN is useful for making candidate controllers compared with NN.

In this paper, we expand candidate controllers to cover the range of the several pole lengths and clarify that MNN is also useful for making expanded candidate controllers.

### 1 Introduction

Multiple control system which consists of a family of candidate controllers, a supervisor and a plant to be controlled is a kind of adaptive control system[4]. The basic idea of multiple control system is as follows, (1)Dividing the environmental conditions into several conditions, (2)Generating candidate controllers corresponds to each conditions, (3)Switching candidate controllers by a supervisor.

There are two types of switching methods for multiple controller. One is based on monitoring estimated plant parameter values[5]. The other is based on identification errors between the plant outputs and the identification model outputs[6].

One of the serious problems in these methods is that their performance may degrade, and what is worse, systems may lose their stability in case when the enough accuracy of plant parameter estimators or multiple identification model can not be achieved.

To solve this problem, one of the co-authors proposed a construction method for logic-based switch-

ing control system which does not have any model to identify environmental conditions[2]. Multilayer neural network(NN) [7] is used for making candidate controllers in this system.

We proposed multi-modal neural network(MNN) to enhance candidate controllers[1]. We applied MNN to candidate controllers for cart-pole balancing system[3]. In this system, the change in pole length corresponds to the change in environmental condition. We generated candidate controllers which were designed for 0.5[m] pole and confirmed the efficiency of MNN compared with NN.

In this paper, we attempt to make candidate controllers to cover the pole length of 0.4, 0.5 and 0.6[m]. We generate two kinds of candidate controllers to compare NN with MNN and show that MNN is more useful for making expanded candidate controllers than the conventional NN.

### 2 Multi-modal Neural Network

Usual neural networks(NN) is used for nonlinear controller[2] because NN can map a given nonlinear function by its nonlinear characteristics. But, it is difficult for usual NN to learn complicated function in enough accuracy [8]. As a function is more complicated, the more synapse and more learning time are

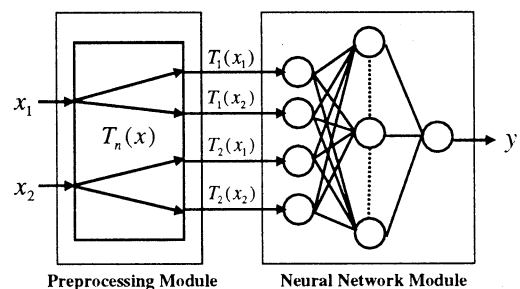


Fig.1. Structure of Multi-modal Neural Network.

needed.

We propose a multi-modal Neural Network in order to improve NN's nonlinear characteristics. The MNN consists of preprocessing module and postprocessing module(NN module)(Fig.1). The preprocessing module transforms input signals into spectra by Chebyshev polynomials. Chebyshev polynomial is defined as follows.

$$\begin{aligned} T_n(x) &= \cos(n \cdot \cos^{-1} x) \\ T_0(x) &= 1, T_1(x) = x, T_2(x) = 2x^2 - 1, \dots \end{aligned} \quad (1)$$

The outputs of preprocessing module are used as inputs of the following NN module.

The preprocessing module cause enhancement of NN's nonlinear characteristics. We validated the effectiveness of the MNN in various experiments[1].

We expect that an improvement of nonlinear property of neural network enables candidate controllers to control a complex system.

### 3 Multiple Control System Using MNN

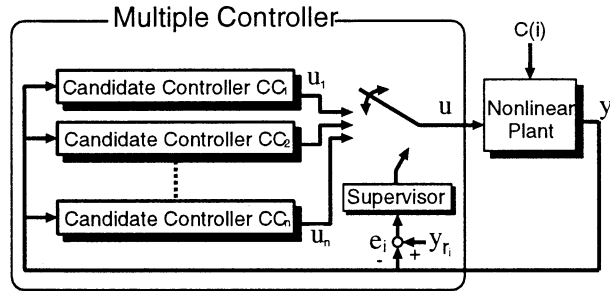


Fig.2. Multiple Control System.

#### 3.1 Multiple Control System

Multiple control system is a kind of adaptive control systems for nonlinear systems whose dynamic characteristic and environmental conditions change widely. A multiple controller consists of a supervisor and candidate controllers. The supervisor switches candidate controllers one by one according to the change of environmental condition.

We focus on a multiple control system which does not have any model to identify environmental conditions[2]. A schematic figure of the multiple control system is shown in Fig.2. In this system, candidate controllers are composed of multilayer neural networks. A candidate controller has to be able to

control a nonlinear plant under some environmental conditions.

We proposed the generating method of candidate controllers for an environmental condition[3]. In this paper, we expand the generating method of candidate controllers which is usable for several environmental conditions.

#### 3.2 How to Generate Candidate Controllers

We need to perform unsupervised learning to generate candidate controllers because we do not have training data. The values of the synapse weights are optimized by Genetic Algorithm(GA)[9]. An individual corresponds to a candidate controller. The procedure of GA is as follows.

- (a) Initialization
- (b) Evaluation
- (c) Judgement of Termination
- (d) Selection
- (e) Crossover
- (f) Mutation

The computation is repeated from (b) to (f) until terminal condition is satisfied.

In the evaluation process, local fitness of candidate controller is calculated by controlling the plant(Fig.3). Local fitness means a fitness under an environmental condition. When candidate controllers are designed for controlling the plant under n environmental conditions, the total fitness is derived from following equation.

$$\begin{aligned} fitness(total) &= fitness(local_1) + \\ &fitness(local_2) + \dots + fitness(local_n). \end{aligned} \quad (2)$$

This method enables us to generate candidate controllers for several environmental conditions.

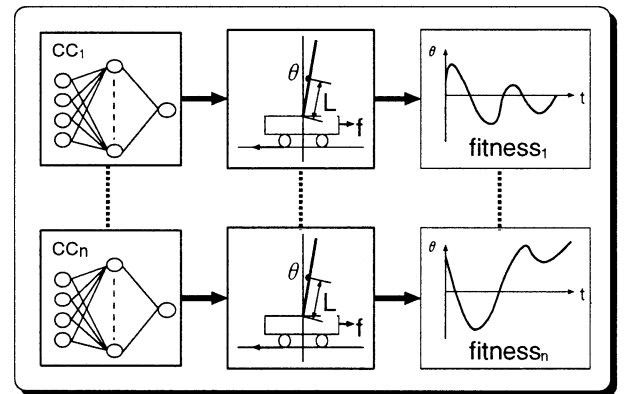
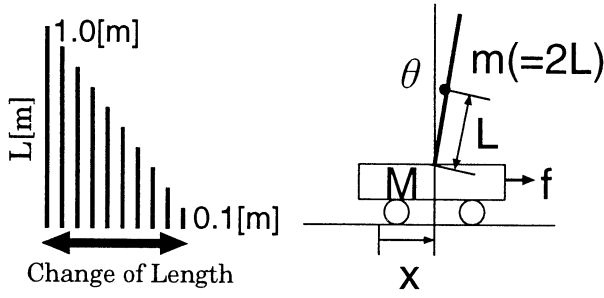


Fig.3. Evaluation of controllers.



- $L$  : Half Length of Pole  
 $m$  : Mass of Pole(=  $2L$ )  
 $M$  : Mass of Cart  
 $\theta$  : Angle of Pole from Vertical  
 $x$  : Position of Cart on Track  
 $f$  : Force Applied to Cart

Fig.4. Cart-Pole Balancing System.

## 4 Cart-Pole Balancing Problem

We attempt to make candidate controllers to control cart-pole balancing system. In this system, the change in pole length corresponds to the change in environmental condition.

### 4.1 Simulation conditions

A cart is free to move along a one-dimensional track and a pole is only free to rotate in the vertical plane of the cart and track(Fig.4). The movement of the pole is described by the following equations:

$$\begin{bmatrix} 4/3mL^2 & mL\cos\theta \\ mL\cos\theta & m+M \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{x} \end{bmatrix} + \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{x} \end{bmatrix} - \begin{bmatrix} mLg\sin\theta \\ mL\theta^2\sin\theta \end{bmatrix} = \begin{bmatrix} 0 \\ f \end{bmatrix}. \quad (3)$$

The fourth-order Runge-Kutta method with a step size of 0.01s is used to compute  $\theta, \dot{\theta}, x$  and  $\dot{x}$ . The output of the neural network is vary between  $-1.0$  and  $1.0$ , where  $-1.0$  represents a pushing force of 50(N) to the right.

The objective of the control is to balance the system such that the pole is balanced in a vertical position, simultaneously not falling beyond predefined

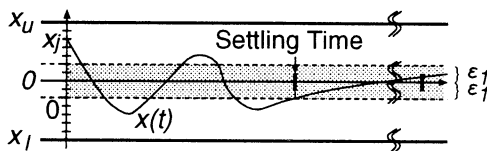


Fig.5. The definition of the settling time.

vertical angles  $\pm 15(\text{deg})$  and the cart remains within the bounds of the horizontal track ( $\pm 1$  meter from the center).

We define “settle” as the condition which satisfied  $|\theta| < 1.0 \times 10^{-3}$  and  $|\dot{\theta}| < 1.0 \times 10^{-3}$ . Settling time(s) means a period from the start of control to the state of settling(Fig.5). The pole length changes with time from 0.1(m) to 1.0(m) by 0.1(m). It is assumed that dynamics of the system and length of the pole are unknown. Initial states are  $\theta = 0, \pm 1, \pm 2, \pm 3(\text{deg}), \dot{\theta} = 0, \pm 3, \pm 6, \pm 9(\text{deg/s}), x = 0(\text{m})$  and  $\dot{x} = 0(\text{m/s})$ .

In the previous report, we generated candidate controllers for an environmental condition( $l = 0.5[\text{m}]$ ) and confirmed that MNN can settle the pole in shorter time than NN[3]. We generate expanded candidate controllers in the next section.

### 4.2 Simulation Results

We generate two kinds of candidate controllers to cover the pole length of 0.4, 0.5 and 0.6[m](Fig.6).

- (a)Candidate controllers based on NN  
 (b)Candidate controllers based on MNN

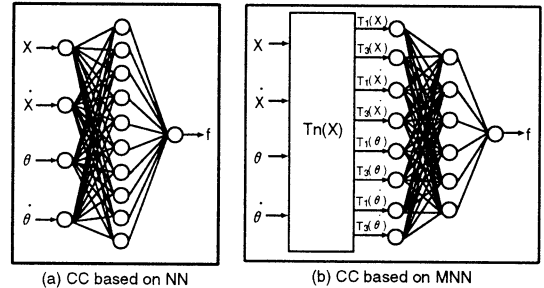


Fig.6. The structure of candidate controllers.

Tab.1 shows the structure of neural network and the number of synapse weight. Chebyshev polynomials which have terms of degree one and three are used as preprocessing module in MNN. The number of MNN's synapse weights is less than that of NN's. The number of generation is same.

Tab.1. The structure of neural networks.

	Net structure	No. of synapse
NN	4-10-1	66
MNN	8-6-1	61

We have five trials starting with different seeds of random number. The elite's settling time of NN and MNN is shown in Tab.2-4. In these tables, boldface means the settling time of a controller's is shorter than

that of the other's in the same column. The shorter settling time implies better controller.

The settling time of MNN is longer than that of NN at trial number 2 and 5. But except for these cases, the minimum settling time of MNN is shorter than that of NN.

Moreover, we design the candidate controllers which can settle the pole within 10(s). The MNN controllers satisfy the condition in all cases. But the NN controllers can not satisfy the condition at trial number 3 and 4 in all the pole length.

These results imply that MNN is superior to NN in unsupervised learning. The reason why MNN is superior to NN is as follows. MNN divides the state space nonlinearly, but NN merely divides the state space linearly.

Tab.2. The settling time for 0.4[m] pole.

Trial Number	1	2	3	4	5
NN	9.55	<b>8.03</b>	15.75	15.47	<b>7.98</b>
MNN	<b>6.66</b>	9.09	<b>7.13</b>	<b>7.44</b>	9.53

Tab.3. The settling time for 0.5[m] pole.

Trial Number	1	2	3	4	5
NN	9.55	<b>8.03</b>	15.75	15.47	<b>7.98</b>
MNN	<b>6.66</b>	9.09	<b>7.13</b>	<b>7.44</b>	9.53

Tab.4. The settling time for 0.6[m] pole.

Trial Number	1	2	3	4	5
NN	8.48	<b>3.77</b>	12.91	13.83	<b>4.40</b>
MNN	<b>3.77</b>	8.17	<b>4.31</b>	<b>4.11</b>	7.60

## 5 Conclusion

We focused on particular multiple control system[2] and improved nonlinear characteristics of candidate controllers under an environmental condition[3].

We expanded candidate controllers to be able to control several environmental conditions. We generated candidate controllers for cart-pole balancing system and clarified MNN is more useful than conventional NN for making candidate controllers.

The future work is the construction of multiple controller and the examination of its performance.

## Acknowledgement

This research is supported by the East JR Endowed Chair of Large Scale System Stages Engineering in Tohoku University. The authors would like to address thankfulness to East JR Co.

## References

- [1] I. Yoshihara, T. Nakagawa and M. Yasunaga, "A multi-modal neural network using Chebyshev polynomials", *Proceedings of the 13th KACC*, pp. 250-253, 1998.
- [2] M. Inaba, I. Yoshihara, H. J. Guo, K. Nakao, K. Abe, "A proposal of switching control system based on speculative control and its application to antiskid braking system", *Proceedings of the 12th KACC*, pp. 585-588, 1997.
- [3] T. Nakagawa, M. Inaba, K. Sugawara, I. Yoshihara and K. Abe, "A proposal of enhanced neural network controllers for multiple control systems", *Proceedings of the 13th KACC*, pp. 201-204, 1998.
- [4] A. S. Morse, "Control Using Logic-Based Switching", Alberto Isidori, editor, *Trends in Control*, Springer, pp. 69-113, 1995.
- [5] N. Satake, O. Ito, K. Kobayashi and O. Yagishita, "Cooperation Control of Wet Pump by Fuzzy Adaptive Controller", *Trans. IEE Japan*, **109-C**, no.5, pp. 361-366, 1989(in Japanese).
- [6] K. S. Narendra and J. Balakrishnan, "Adaptive Control Using Switching and Tuning", *Proc. of the Eighth Yale Workshop on Adaptive And Learning Systems*, pp. 13-15, 1994.
- [7] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning internal representations by error propagation", D.E.Rumelhart and J.L.McCelland, editors, *Parallel Distributed Processing*, Cambridge, MA:MIT Press, pp. 318-362, 1986.
- [8] A. Nabatame and N. Ueda "Object Recognition Using Chebychev Networks", *Transactions of Information Processing Society of Japan*, pp. 1542-1551, 1991(in Japanese).
- [9] J. H. Holland, "Adaptation in Natural and Artificial Systems", *The Univ. Michigan Press*, MIT Press, 1975.

# Dynamical Recognition via Hybrid Neural Networks

N. Honma  
Col. of Medical Sciences  
Tohoku University, Sendai 980-8575, JAPAN  
(honma@abe.ecei.tohoku.ac.jp)

K. Abe  
Graduate School of Eng.

H. Takeda  
Faculty of Eng.  
Tohoku Gakuin University  
Tagajyo 985-8537, JAPAN

## Abstract

We propose a new recognition mechanism via hybrid neural networks with feedforward and recurrent connections. An essential point of our mechanism is a *dynamical* recognition based on chaotic EEG activities of mammalian brains. The chaotic activities are generated by designing the recurrent connection weights in our networks. Harnessing the chaotic dynamics of recurrent networks, our hybrid networks can recognize the "known" patterns and their *neighbors* as the conventional recognition methods can. In addition, our networks are able to decide whether input patterns are "known" or "unknown" by observing temporal stability of output patterns.

**Key words:** *Neural networks, Recognition, Chaos, Complex systems.*

## 1 Introduction

Various types of neural networks for character recognition have been proposed and discussed. Most of these networks can be classified into two types: back-propagation networks[1] and associative memory networks with symmetric recurrent connections[2, 3, 4]. Both types are based on the same manner using a *stable* classification boundary, where for associative memory type "stable" means that the dynamics of Hopfield's model is very simple: a "basin", which is a set of input patterns converging to a memorized pattern, is *stable*(fixed). Using this boundary, they can recognize *neighbor* patterns, which have several pixels different from memorized patterns. That is, the recognition of neighbors requires to regard unknown patterns as one of known(memorized) patterns.

On the other hand, Freeman reported that mammalian recognition activities may employ a chaotic dynamics[5]: a) EEG activities without stimulation(no input) are chaotic, b) activities are limit cycles for

known inputs, and c) for *unknown* inputs, activities are chaotic ones different from that of a).

In this paper we propose a new type of recognition network model with chaotic dynamics based on above experiential results. Recognition abilities of our networks are investigated by simulation studies. From the simulation results, we discuss on the network performances: learning of patterns to be memorized, recognition of known(memorized) inputs and their *neighbors*, and *recognition* of "unknown" inputs.

## 2 Hybrid neural networks

### 2.1 Structure of proposed networks

We propose a hybrid neural networks with feedforward and recurrent connections. The connection structure of proposed networks is basically feedforward: the networks have an input, several hidden, and an output layers, where neurons in the input layer have recurrent connections with each other(see figure 1).

Letting  $I_i$  denote a pixel value of an external input pattern,  $I_i$  is fed to the  $i$ -th neuron in the input layer, and conveyed to the output layer through the hidden layer as follows:

$$x_i(t+1) = G \left( \sum_j u_{ij} x_j(t) + s I_i(t) + \theta_i(t) \right) \quad (1)$$

$$h_k(t) = G \left( \sum_i v_{ki} x_i(t) + \phi_k(t) \right), \quad (2)$$

$$o_l(t) = G \left( \sum_k w_{lk} h_k(t) + \Psi_l(t) \right), \quad (3)$$

$$G(z) = \frac{1}{1 + \exp \left( -\frac{z}{a} \right)}, \quad (4)$$

$t = 1, 2, \dots$ . Here,  $s$  is a weight coefficient of external inputs, and  $a$  is a constant of the function  $G(z)$ .  $x_i, h_k$

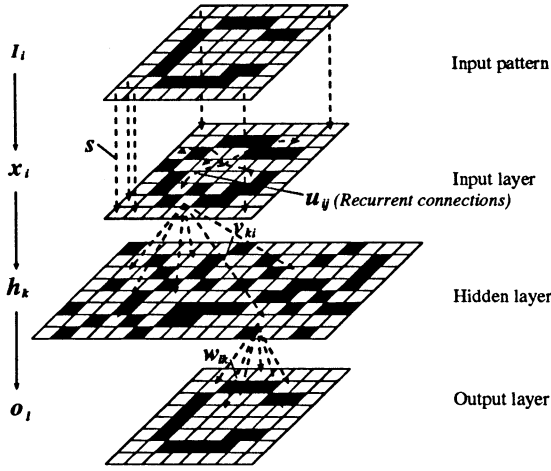


Figure 1: The hybrid structure of the proposed neural network.

and  $o_l$  are outputs of neurons,  $u_{ij}$  are recurrent connection weights,  $v_{ki}$  and  $w_{lk}$  are feedforward weights, and  $\theta_i$ ,  $\phi_k$  and  $\Psi_l$  are thresholds, respectively.

## 2.2 Back-propagation learning

While a character pattern continue to be fed to our networks for a period, the *static* time series of the input pattern is transformed into a *dynamic* series of outputs of recurrent neurons, and it is conveyed to the feedforward layers. The output series of the recurrent neurons depends on the character patterns. That is, regarding each of the outputs as a state variable, an attractor on the state-space is also changed according to the character patterns[7].

Parameters of feedforward networks,  $v_{ki}$ ,  $w_{lk}$ ,  $\phi_k$  and  $\Psi_l$ , are changed by the back-propagation algorithm[6] in order that the output pattern of the networks and the input pattern may be the same.

## 2.3 Design of dynamical complexity

Our design strategy of recurrent connections is based on the experiential results of EEG activities[5]. While many measurements of dynamical complexities have been proposed, in this study we employ the largest Lyapunov exponent  $\lambda$ [8].

When we design the largest Lyapunov exponent to be much greater than 0, the strange attractors of the chaotic dynamics will be excessively complex; the back-propagation learning of character patterns is too difficult, or is computational expensive since mapping

functions of feedforward networks are required to be excessively complicated nonlinear functions. In addition, the rate of correct recognition for memorized patterns and their *neighbors* will be worse than that of networks with simple attractors(dynamics).

On the other hand, if we design the Lyapunov exponent of the networks to be much less than 0, it is impossible to realize a recognition mechanism based on the chaotic dynamics. In fact, these behaviors are suggested qualitatively by the simple simulation studies. We design, therefore, the exponent to be 0 or approximately ( $\lambda \simeq 0$ : the edge of chaos), in order that the networks may have a high rate of correct recognition for memorized patterns and that the dynamics may be chaotic for the no input case.

The dynamics of recurrent networks are designed by a qualitative method proposed in our previous studies[9]. An essential core of this design method is as follows. At the first, we introduced a new parameter  $\overline{\sigma^2}$  given as

$$\overline{\sigma^2} = \frac{1}{12N} \sum_{ij} u_{ij}^2. \quad (5)$$

We chose random numbers for connection weights of the networks  $u_{ij}$  in order to initialize the parameters, and designed thresholds of the neurons  $\theta_i$  by the following equation:

$$\theta_i = -\frac{1}{2} \sum_j u_{ij}. \quad (6)$$

By simulation studies, we had a qualitative relation between the largest Lyapunov exponent  $\lambda$  and the new parameter  $\overline{\sigma^2}$  for large-scale networks which composed of numerous neurons [9](see figure 2).

From figure 2, we can determine the parameter value of  $\overline{\sigma^2} = \sigma_s^2$  corresponding to a desired value of the exponent  $\lambda$ . Finally,  $u_{ij}$  and  $\theta_i$  are designed by as follows.

**Step 1:** Choose random numbers for all  $u_{ij}$ .

**Step 2:** Calculate  $\overline{\sigma^2}$  by initialized  $u_{ij}$ , then evolve  $u_{ij}$  such that  $\overline{\sigma^2} = \sigma_s^2$ .

$$u_{ij} \leftarrow \sqrt{\frac{\sigma_s^2}{\overline{\sigma^2}}} \times u_{ij}. \quad (7)$$

**Step 3:** Calculate  $\theta_i$  by (6).

## 3 Dynamical recognition

We have tested performances of our networks for several recognition tasks. Input patterns are four

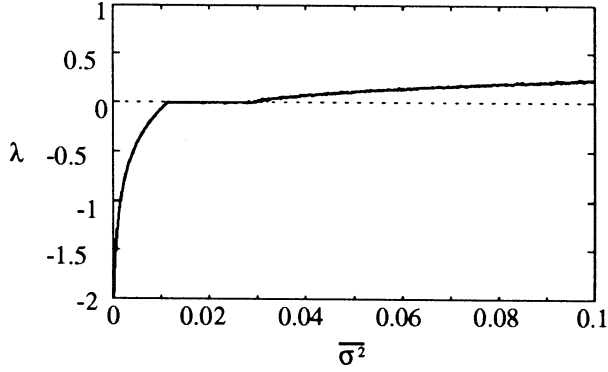


Figure 2: The largest Lyapunov exponent  $\lambda$  as a function of the new parameter  $\overline{\sigma^2}$  under the condition  $\theta_i = -\frac{1}{2} \sum_j u_{ij}$  for large-scale networks.

characters "C", "A", "S" and "E", consisting of  $64 (8 \times 8)$  pixels (see figure 3). These character patterns are memorized by the back-propagation learning described in Sec.2.2 for 100 steps ( $t = 0, 1, \dots, 99$ ). The weight of external inputs  $s$  is 0.04, and the networks have 64-128-64 neurons in the input-hidden-output layers, respectively.

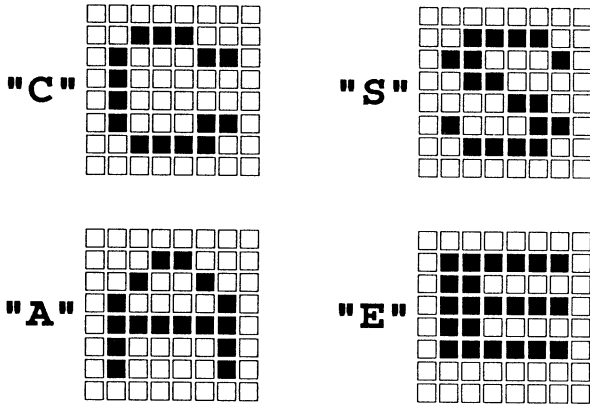


Figure 3: Character patterns as learning objects.

### 3.1 Basic performance

#### 3.1.1 No input case

Once learning has finished, the output series of our network depends on the initial states of recurrent neurons when no external input is fed to the network. Now how do we set the initial values for the no input case?

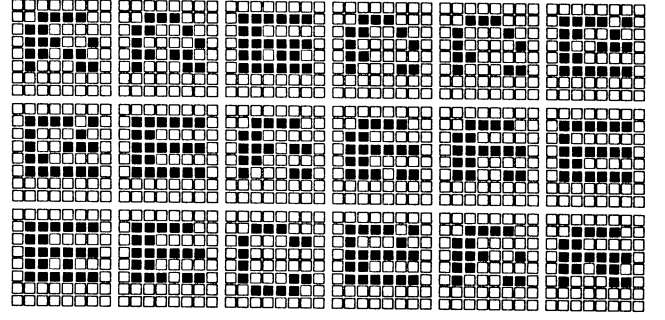


Figure 4: Temporal variations of output patterns for the no input case.

A random value is not adequate to simulate an activity without stimulation since corresponding mammalian dynamics may not be random activities. When we give an input pattern to the networks then stop to give the input at a time, after this stop, state values of neurons will be more suitable for simulating the no input case.

In this simulation, we stopped to give a pattern at a time  $t = 5$  and we observed behaviors after the stop. Figure 4 shows a part of temporal variations of output patterns after  $t = 5$ . Note that for the no input situation, our network activities are chaotic: output patterns are wandering about various patterns.

#### 3.1.2 Known input case

When four memorized patterns are fed to our networks, the network outputs are all correct patterns for 100 steps. This suggests that an appropriate function defined by feedforward weights is determined by learning of target limit cycles which are generated by recurrent networks.

### 3.2 Generalization ability

Figure 5 shows temporal changes of output patterns for a neighbor input pattern which has three reversed pixels to the known pattern "A". The number of correct recognition is 91 for 100 steps and less than that of the non-reversed input case. Other patterns of the output "A", however, are *neighbor* ones of "A" and not the rest of other known patterns, "C", "S" and "E". This means that *dynamical* patterns forced by *neighbors* of the known input "A" are close to ones forced by the known input. From this result we can expect that proposed networks are able to have generalization ability as previous models do.



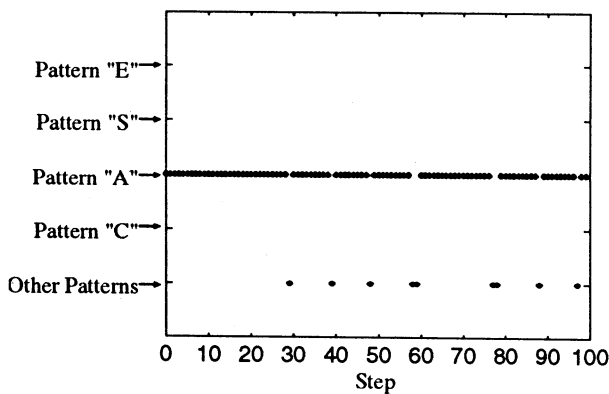


Figure 5: Temporal changes of output patterns for an input pattern which has three reversed pixels to the known pattern "A".

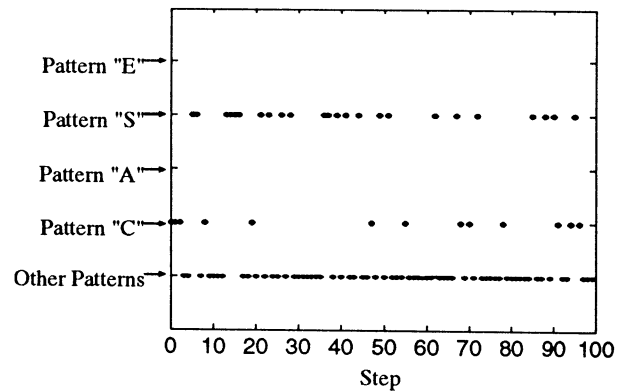


Figure 6: Temporal changes of output patterns for an unknown input pattern "B".

### 3.3 Recognition of *unknown* inputs

Figure 6 shows temporal changes of output patterns for an unknown input pattern "B". Note that the number of incorrect recognition, i.e., regarding the unknown input "B" as one of known patterns is very few. Other patterns include many patterns different from each other. In addition, temporal changes of output patterns are chaotic. This activity is corresponding to Freeman's result for unknown inputs described in Sec.1, namely the temporal changes for unknown inputs are different from that of the known inputs case. The networks, therefore, can decide whether input patterns are "known" or "unknown" by observing temporal stability of output patterns.

## 4 Conclusions

We have proposed new type of networks with a chaotic dynamical recognition mechanism based on the experiential results of mammalian EEG activities. Simulation results show that our networks can recognize known patterns and their *neighbors*. In addition, the networks can decide whether input patterns are known or unknown. This recognition of "unknown" patterns will make it possible the networks to learn the new patterns automatically.

## References

- [1] D.E.Rumelhart, G.E.Hinton, R.J.Williams(1986), Learning Internal Representations by Error Propagation, *Parallel Distributed Processing*, The MIT Press, 1, pp.318-362
- [2] K. Nakano(1972), Associatron - A model of associative memory, *IEEE Trans.*, **SMC-2**, pp.380-388
- [3] T. Kohonen(1972), Correlation matrix memories, *IEEE Trans.*, **C-21**, pp.353-359
- [4] J. A. Anderson(1972), A simple neural network generating an interactive memory, *Math. Biosciences*, **14**, pp.197-220
- [5] C. A. Skarda, W. J. Freeman(1987), How brains make chaos in order to make sense of the world, *Behavioral and Brain Sciences*, **10**, pp.161-195
- [6] D.E.Rumelhart, G.E.Hinton, R.J.Williams(1986), Learning representations by back-propagating errors, *Nature*, **323**, pp.533-536
- [7] K. Gohara(1995), Neurodynamics Excited by the Spatio-Temporal Patterns: On the Excited Attractor Model(in Japanese), Technical Report of IEICE, NC94-153, pp.299-306
- [8] A. Wolf, J. B. Swift, H. L. Swinney and J. A. Vastano(1989), Determining Lyapunov Exponents from a time series, *Physica*, **16D**, pp.285-315
- [9] N.Honma, K.Kitagawa, K.Abe, H.Takeda(1997), An Autonomous Criterion of Learning Methods for Recurrent Neural Networks, *Proc. of the 2nd ASCC*, **II**, pp.219-222

## Incremental Evolution of Neural Controllers for Navigation in a 6-legged Robot

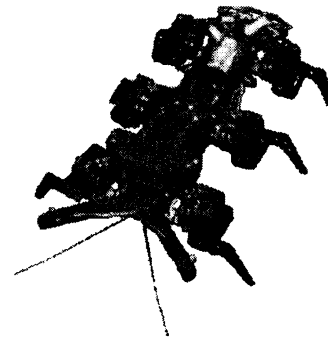
D. Filliat

J. Kodjabachian  
AnimatLab / OASIS - LIP6  
Paris. France.

J.-A. Meyer

### Abstract

This paper describes how the SGOCE paradigm has been used within the context of a "minimal simulation" strategy to evolve neural networks controlling locomotion and obstacle-avoidance in a 6-legged robot. Such controllers have been first evolved through simulation and then successfully downloaded on the real robot.



## 1 Introduction

In two previous articles (Kodjabachian and Meyer [5], Kodjabachian and Meyer [6]), it has been shown how the so-called SGOCE evolutionary paradigm could be used to generate neural networks capable of controlling the navigation of an artificial insect. More specifically, developmental programs generating controllers for locomotion, obstacle-avoidance and gradient-following have been automatically generated, thus endowing the insect with navigation abilities through a simple guidance strategy (Trullier and Meyer [10], Trullier et al. [11]). In this paper, we show how the SGOCE paradigm has been used to generate neural controllers for locomotion and obstacle-avoidance in a real 6-legged SECT robot manufactured by Applied AI Systems (figure 1). Results obtained on gradient-following will be published elsewhere. A review of similar approaches involving a variety of neural controllers and robots is available in Meyer [8].

## 2 The SGOCE evolutionary paradigm

The SGOCE evolutionary paradigm is characterized by an encoding scheme, by an evolutionary algorithm, by an incremental strategy, and by a fitness evaluation procedure that will be sketched in turn. More detailed descriptions can be found in Chavas et al. [1], Filliat [2], Kodjabachian and Meyer [5], Kodjabachian and Meyer [6].

Figure 1: The SECT robot. It is equipped with infrared sensors that can be used to detect obstacles, and with light sensors that can be used for light-following. Each leg is controlled by two servo-motors that react to angular-position commands, e.g., one for horizontal moves and one for vertical moves.

### 2.1 Encoding scheme

The encoding scheme of SGOCE (figure 2) is a geometry-oriented variation of Gruau's cellular encoding (Gruau [3]). The developmental programs that are evolved have a tree-like structure and call upon developmental instructions that cause a set of precursor cells positioned by the experimenter in a 2D metric substrate to divide, die, or grow efferent or afferent connections. In particular, such cells can get connected to each other, or to sensory cells or motoneurons that have also been positioned in the substrate. Thus, a possibly short and compact genotype may ultimately produce a complex phenotype, i.e., a fully recurrent neural network made of individual leaky-integrator neurons and able to control the behavior of the robot through its sensors and actuators.

### 2.2 Evolutionary algorithm

The evolutionary algorithm of SGOCE is a steady-state genetic algorithm that involves a population of well-formed developmental programs whose structure

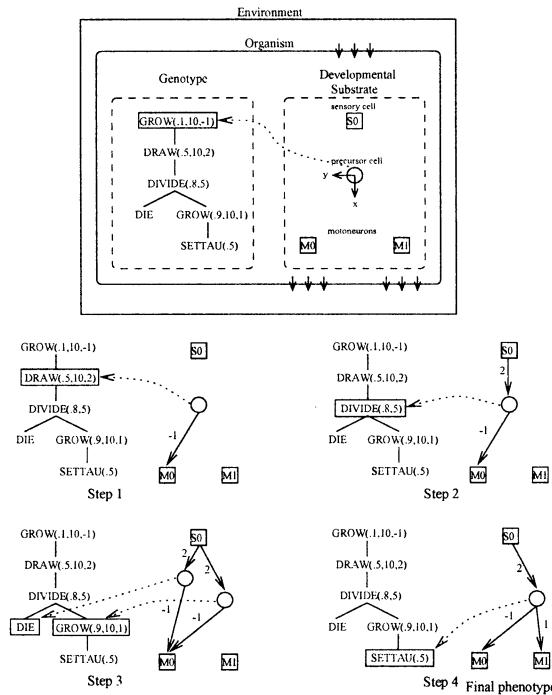


Figure 2: The developmental encoding scheme of SGOCE. The genotype that specifies the robot's nervous system is encoded as a program whose nodes are specific developmental instructions. This developmental program is read at a different position by each cell in the substrate. The precursor cell first makes connections with the motoneuron M0 and the sensory cell S0 (Steps 1 and 2). Then it divides, giving birth to a new cell that gets the same connections than those of the mother cell (Step 3). Finally, the mother cell dies, while the daughter cell makes a connection with the motoneuron M1 and changes the value of its time constant (Step 4).

is constrained by a grammar provided by the experimenter (figure 3). The use of such a grammar makes it possible to reduce the size of the genotypic space explored by the algorithm and to limit the complexity of the neural networks that are evolved.

### 2.3 Incremental strategy

Finally, the SGOCE paradigm resorts to an incremental strategy that takes advantage of the geometrical nature of the developmental process. In particular, it makes it possible to automatically generate appropriate controllers and behaviors through successive stages, in which good solutions to a simpler version of a given problem are iteratively used to seed the initial

<b>Terminal symbols</b>	
DIVIDE, GROW, DRAW, SETBIAS, SETTAU, DIE, NOLINK, DEFBIAS, DEFTAU, SIMULT3, SIMULT4.	
<b>Variables</b>	
Start1, Level1, Level2, Neuron, Bias, Tau, Connex, Link.	
<b>Production rules</b>	
Start1	→ DIVIDE(Level1, Level1)
Level1	→ DIVIDE(Level2, Level2)
Level2	→ DIVIDE(Neuron, Neuron)
Neuron	→ SIMULT3(Bias, Tau, Connex)   DIE
Bias	→ SETBIAS   DEFBIAS
Tau	→ SETTAU   DEFTAU
Connex	→ SIMULT4(Link, Link, Link, Link)
Link	→ GROW   DRAW   NOLINK
<b>Starting symbol</b>	
Start1.	

Figure 3: This grammar defines a set of developmental programs, i.e., those that can be generated from it, starting with the Start1 symbol. When this grammar is used, a cell that executes such a program undergoes two division cycles, yielding four daughter cells, which can either die or modify internal parameters (e.g., time constant or bias) that will influence their behavior within the final neural controller. Finally, each surviving cell establishes a number of connections, either with another cell, or with the sensory cells or motoneurons that have been positioned by the experimenter in the developmental substrate. According to this grammar, no more than three successive divisions can occur and the number of connections created by any cell is limited to four. Thus, the final number of interneurons and connections created by a program well-formed according to this grammar cannot be greater than 8 and 32, respectively.

population of solutions likely to solve a harder version of the same problem.

Thus, in a first stage of the present work, the SGOCE paradigm was used to generate a recurrent neural network controlling straight locomotion in the SECT robot. At the end of this stage, this network was frozen, in the sense that the number of its neurons, their individual parameters, and their intra-modular connections were not allowed to evolve anymore. However, during a second evolutionary stage, an additional recurrent neural network was evolved and its neurons were allowed to grow, not only intra-modular connections between themselves, but also inter-modular connections to neurons in the locomotion controller. This additional controller was expected to modulate the leg movements secured by the first controller, so as to make it possible for the robot to turn in the presence of an obstacle in order to avoid it.

Figure 4 describes the two substrates that have been used to generate the two modules of the present application.

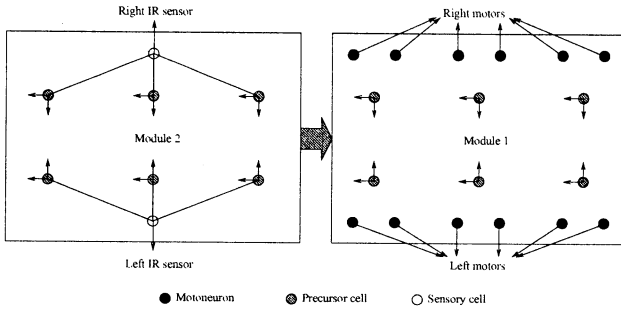


Figure 4: The substrates and the modular approach that have been used in the present work. Module 1 produces straight walking, while Module 2 modifies the behavior of Module 1 in order to avoid obstacles.

## 2.4 Fitness evaluation

Fitness evaluation is one of the main difficulties of the evolutionary design of controllers in real robots (Meyer et al. [9], Mataric and Cliff [7]). Such difficulties are enhanced in the case of legged robots, because they tend to be more brittle than their wheeled counterparts, and because fitness evaluations require a lot of time and cannot be easily automated. We therefore chose to use simulations to assess the fitness of our controllers, taking advantage of an argument put forth by Jakobi (Jakobi [4]), namely that what really matters is to accurately simulate the efficient behaviors that will be used by the real robot. Less efficient behaviors, which an efficient robot won't exhibit in reality, do not need to be minutely simulated, as long as we are sure that their fitnesses will be lower than the fitnesses of the behaviors that are sought. Pushing such reasoning to the extreme, Jakobi evolved neural controllers for an octopod robot capable of walking, of avoiding obstacles using its infra-red sensors, and backing away from objects that were hit with its bumpers. Jakobi's approach didn't resort to any simulation of the robot's behavior in its environment, and only relied on the specification that legs on the floor should move backwards as fast as possible, and that legs in the air should move forward as fast as possible. Given such specification, the simulation only rewarded controllers that did generate these movements. However, despite the practical success of Jakobi's approach, our aim was to provide less constraints on the target behavior. Therefore, we only specified that the robot should go ahead as far as possible while avoiding obstacles, and we didn't provide any hints about leg movements. To this end, we had to design a simulation of the behavior of the robot in its environment.

As Jakobi points out, the difficulty in devising a legged robot simulation is to manage the cases when some leg slippage occurs. However, because such events are only involved in poorly efficient behaviors, they are not expected to occur with a fast-walking gait. As a consequence, controllers producing leg slippage will never be used by an efficient real robot, and therefore leg slippage does not need to be accurately simulated, thus tremendously simplifying the simulation problem.

According to such considerations, our simulation assumed that all the legs were characterized by the same constant slippage factor, and simply calculated the movement of the robot body that minimized the slippage of any leg touching the floor between two time steps. As a consequence, if the real movement did not involve any slippage, the calculated movement was exact and, conversely, if the real movement did involve slippage, the calculated movement was a good approximation of the real one. This computation entailed the zeroing of the partial derivatives, with respect to translation and rotation, of the sum of the squared distances covered by each leg touching the floor. Technically, it only required a linear system inversion, which could be performed very efficiently (Filliat [2]).

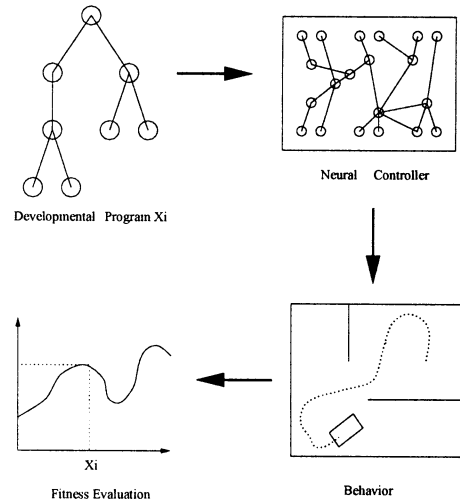


Figure 5: The three stages of the fitness evaluation procedure. An evolved developmental program is executed to yield an artificial neural network. Then, the neural network is used to control the behavior of a simulated robot. Finally, the fitness of the program is assessed according to the result of the simulation.

Such simulations have been used to assess the fitness of each developmental program produced by the evolutionary process (figure 5). Once an efficient con-

troller was thus obtained, it was downloaded on the SECT robot, where its ability to generate the target behaviors in reality was assessed again.

### 3 Experimental results

#### 3.1 Locomotion

The 2D substrate that was used in this experiment is the Module 1 shown in figure 4. It contained 12 motoneurons that were connected to the 12 motors of the robot, in the sense that the activity level of a given motoneuron determined the target angular position that was sent to the corresponding servo-motor. The six precursor cells executed the same evolved developmental program in order to impose symmetrical constraints to the growing neural network. The corresponding fitness was the distance covered in an obstacle-free environment during a fixed amount of time, increased by a slight bonus encouraging any leg motion (Kodjabachian and Meyer [5], Kodjabachian and Meyer [6]). Finally, the size of the population was of 100 individuals that evolved during 500 generations (taking 24 hours on a SUN Ultra 1).

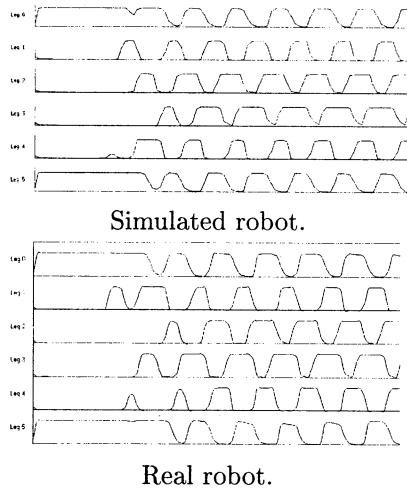


Figure 6: Comparison of leg movements in the simulated and the real robot. Both graphs show commands sent to leg swing motors. Although the controller's outputs might have been altered by the procedure preventing over currents in the real robot, the commands actually sent to the motors, and hence the robot's overall behaviors, are qualitatively the same in simulation and in reality.

All the evolved controllers could be classified into

two categories: those generating tripod gaits and those generating symmetrical gaits (i.e., moving the corresponding legs on both sides of the robot simultaneously). Such controllers were as efficient in reality as they were in simulation (figure 6). The main differences between both situations were due to the fact that, on the real robot, a continuous monitoring of the motor currents might entail modifying the motor commands independently of the neural network when such current were too high. This security procedure, which was implemented to avoid motor breaks in leg-blocking situations, was triggered in a few occasions when symmetric gaits were used, because such gaits occasionally provoked jumps producing very high motor currents.

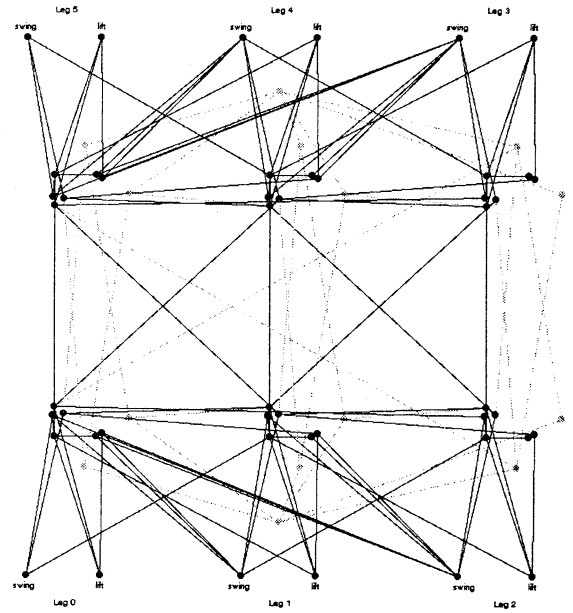


Figure 7: An example of an evolved controller. Black neurons belongs to the module controlling locomotion, gray neurons belong to the module controlling obstacle avoidance. The first controller contains 48 neurons and 82 connections; the second one contains 14 neurons and 36 connections.

Figure 7 shows the best tripod-gait controller that has been obtained. We analyzed its behavior in order to understand how this gait was generated. The mechanism responsible for leg lift is straightforward. It calls upon 6 pattern generators, one for each leg, made up of only 3 neurons. These pattern generators are synchronized by two connections between the legs of the same side of the body, and by two connections linking 2 symmetric legs on each side of the body. The

mechanism producing leg swing is far more intricate. In fact, the activation of a given leg's swing neuron depends on neurons involved in the control of all the other legs, and it cannot be decomposed in six similar units as is the case for the lift neurons.

### 3.2 Obstacle-avoidance

Obstacle-avoidance was sought using a second module whose neurons could be connected to those of the tripod-gait controller of figure 7. The substrate of this second module is shown on figure 4. It contained two input neurons linked to the IR sensors of the robot, and six precursor cells as in the first module. These six cells have forced connections to the input neurons. Each IR sensor was binary, i.e., it returned 0 when it detected no obstacle, and it returned 1 when an obstacle was detected within a 30 cm-range. The robot evolved in a closed environment containing some obstacles (figure 8). Instead of specifying what leg movements should be favored in each possible sensory circumstance, as Jakobi [4] did, the fitness was the distance covered by the robot until it touched an obstacle or until the simulation time was over. The population was made of 100 individuals and evolution lasted 200 generations.

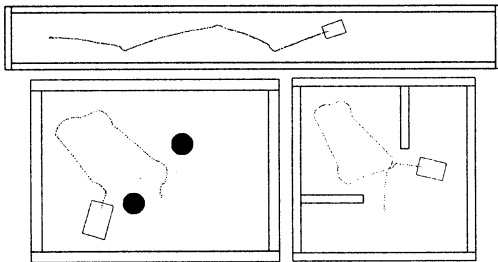
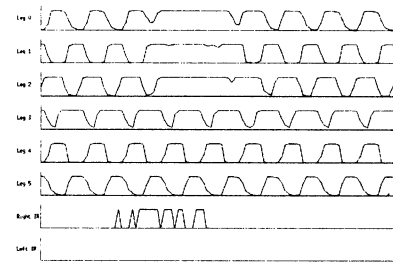
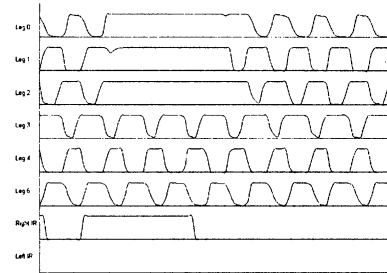


Figure 8: Examples of the behavior of the simulated robot in different environments.

As shown in figure 9, the robot's simulated and actual behaviors were very similar and quite simple: legs on the opposite side of a detected obstacle were blocked, thus causing the robot to change its direction and to avoid the obstacle. Analyzing the inner workings of the corresponding controller (figure 7), it turned out that such behavior was possible due to a strong inhibitory connection between a given sensor and the swing neurons of the legs on the opposite side of the robot.



Simulated robot.



Real robot.

Figure 9: Leg swings in the simulated and the real robot during obstacle avoidance. When an obstacle is detected at the right (Right IR) of the robot, the three legs on its left (Leg0, Leg1, Leg2) are temporarily blocked.

## 4 Discussion

Results that have been shown here demonstrate that the "mimimal simulation" methodology advocated by Jakobi (Jakobi [4]) may be effectively used to evolve non trivial behaviors in a real robot. However, our implementation of this methodology makes it possible to avoid specifying as many details about the dynamics of each effector involved in the production of the sought behavior as Jakobi was committed to do. Actually, we succeeded to evolve locomotion and obstacle-avoidance in a legged robot simply rewarding movement and punishing obstacle-hitting, and such high-level specification is more in the spirit of the automatic generation of behavioral controllers that evolutionary approaches afford than Jakobi's solution is. Nevertheless, it must be emphasized that a cost is associated to such a benefit, namely that of entailing a more detailed simulation than that of Jakobi. In other words, what is gained at the level of the simulation, may be lost at the level of the fitness evaluation and, for obvious lack of hindsight reasons, it is definitely unclear where it is worth devoting more resources in order to evolve any given behavior on a real robot.

Results that have been shown here also demonstrate the effectiveness of the SGOCE evolutionary paradigm. Its potentialities to evolve guidance navigation capacities in a simulated insect have already been exemplified elsewhere (Kodjabachian and Meyer [5], Kodjabachian and Meyer [6]). In the present work such potentialities have been extended to a real robot, although evolving a light-following behavior is still a matter of current research. For the same lack of hindsight reasons as above, it is however still unclear whether each aspect of this paradigm is mandatory, useful, or without any effect at all. One may wonder, for instance, if another evolutionary procedure than a genetic algorithm wouldn't lead to better results, if the developmental instructions or the constraining grammars that have been used here might not have been replaced by others, and if freezing one controller and letting a second one evolve is a better strategy than evolving both controllers at the same time. Concerning the latter point, however, it has been shown on a specific application involving obstacle-avoidance in a Khepera robot that this was not the case (Chavas et al. [1]) but, again, any generalization to other behaviors and other robots would be definitely premature.

## 5 Conclusions

Assuming that an on-board evolution of neural controllers wouldn't have been feasible on a SECT robot because of the excessively high demands that would have been placed on its motors, we called upon a "minimal simulation" approach and upon the SGOCE evolutionary paradigm to evolve tripod-gait locomotion and obstacle-avoidance. Successful neural controllers have been obtained, both in simulation and in reality, and their inner workings have been deciphered. However, it is yet unclear whether every implementation detail that has been used here was mandatory to the present success, nor whether it would be useful in any other application.

## Acknowledgements

The authors express their gratitude to Dr. Takashi Gomi and to Applied AI Systems for having kindly lent us the SECT robot.

## References

- [1] Chavas, J., Corne, C., Horvai, P., Kodjabachian, J. and Meyer, J.A. "Incremental Evolution of Neural Controllers for Robust Obstacle-Avoidance in Khepera." In Husbands, P. and Meyer, J.A. (Eds.). *Proceedings of The First European Workshop on Evolutionary Robotics - EvoRobot'98*, Springer Verlag, 1998.
- [2] Filliat, D. "Evolution de réseaux de neurones pour le contrôle d'un robot hexapode." *Lip6 Technical Report*, 1998.
- [3] Gruau, F. "Automatic definition of modular neural networks." *Adaptive Behavior*, 3, 1994.
- [4] Jakobi, N. "Running across the reality gap : octopod locomotion evolved in minimal simulation." In Husbands, P. and Meyer, J.A. (Eds.). *Proceedings of The First European Workshop on Evolutionary Robotics - EvoRobot'98*, Springer Verlag, 1998.
- [5] Kodjabachian, J. and Meyer, J.A. "Evolution and Development of Modular Control Architectures for 1-D Locomotion in Six-legged Animats." *Connection Science*, 10, 1998.
- [6] Kodjabachian, J. and Meyer, J.A. "Evolution and Development of Neural Controllers for Locomotion, Gradient-Following, and Obstacle-Avoidance in Artificial Insects." *IEEE Transactions on Neural Networks*, 9:5, 1998.
- [7] Mataric, M. and Cliff, D. "Challenges in evolving controllers for physical robots." *Robotics and Autonomous Systems*, 19(1) , 1996.
- [8] Meyer, J.A. "Evolutionary approaches to neural control in mobile robots." In *Proceedings of the IEEE International Conference on Systems, and Cybernetics*, San Diego, 1998.
- [9] Meyer, J.A., Husbands, P. and Harvey, I. "Evolutionary Robotics: a Survey of Applications and Problems." In Husbands, P. and Meyer, J.A. (Eds.). *Proceedings of The First European Workshop on Evolutionary Robotics - EvoRobot'98*, Springer Verlag, 1998.
- [10] Trullier, O. and Meyer, J.A. "Biomimetic Navigation Models and Strategies in Animats." *AI Communications*, 10, 1997.
- [11] Trullier, O., Wiener, S., Berthoz, A. and Meyer, J.A. "Biologically- based artificial navigation systems: Review and Prospects." *Progress in Neurobiology*, 51, 1997.

# Pulse Neural Network applied to the Binding Problem

— Implementation of decision making of the mobile robot —

\*Masaomi KOJIMA, Akihiro YAMAGUCHI, Masao KUBO, Sadayoshi MIKAMI, Mitsuo WADA,  
Hokkaido University

**Key Words:** Pulse Neural Network, spiking neuron, binding problem, recognition

## Abstract

We deal with Pulse Neural Network (PNN): a neural network in which information is encoded as delays of firing neurons. We present an example of solution of the binding problem that is applied to decision making of mobile robot.

## 1. Introduction

In a high level system such as a brain of human, there exists so-called a 'Binding Problem'<sup>(1)</sup>. This is unavoidable problem on recognizing objects of human or robots. For example, it should arise in the method in which robot distinguish obstacle and a target of a task when robot sees two kinds of objects.

Various solutions for the Binding Problem have been proposed. Amongst them, a solution using temporal correlation is thought to be adequate at present<sup>(2)</sup>.

The neural networks implementing this method is known as a "Pulse Neural Network"<sup>(3)</sup>. In this network, information is encoded as a timing of firing of a neuron not a rate of firing.

In our research, we propose an architecture for Pulse Neural Network aiming at the decision making of the mobile robot in the environment included binding problem inside. We define the binding problem on robots in a computer simulation and verify it with an experiment. We show that the size of pulse neural network can be smaller than that of conventional rate coding neural networks.

In the chapter 2, we explain the binding problem. In the chapter 3, the neuron model we use in this paper is explained. Chapter 4 describes a solution for binding problem by temporal coding with our neuron model. Finally in the chapter 5, we show the setting of our computer simulation and the results.

## 2. Binding Problem

### 2.1 Binding Problem

Binding problem is thought as one of the most important problem in brain.

The binding problem is easily explained by using the example of binding in visual system. The brain of human divides cognitive object into elements of optical feature. In other words, each neuron that represents each element exists in different area. In order to

recognize object properly, it must integrate elements of visual feature. The binding problem is the problem to integrate these divided elements into groups of elements so that the grouped elements represent the result of recognition. Today two approaches are proposed that can solve binding problem in visual system. The first approach is that each element of visual feature represented in various region is integrated finally in their own regions in brain. But in this approach it may cause 'super-positional catastrophe', that huge number of neurons are necessary.

In order to avoid this disadvantage, the second approach has been proposed. This approach is to let elements for features interlace with each other to compose objects. In this case it is an important problem how it interlaces with each other. Some methods are proposed for this problem, within it the method using temporal correlation amongst neurons gets attention<sup>(1)</sup>.

### 2.2 Binding Problem on making decision of robots

In our work, we design the setting as described below, and discuss the binding problem that arises on making decision of robots. We focus on the binding problem on a visual system and explain it based on the situation implemented for making decision of robots.

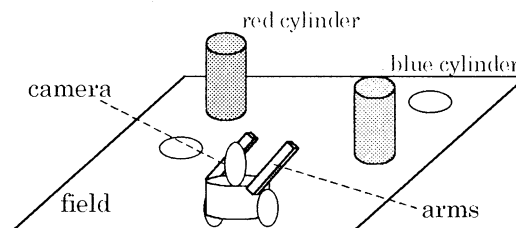


Fig.1. Simulation environment

In the environment, a mobile robot with a camera and arms is placed. There are fixed four points; a red object and a blue cylinder are placed on one of these four points (Fig.1). The place of each cylinder changes at every trial. Two cylinders are not permitted to be placed on the same point.

The task of the mobile robot is to grab a target. In our experiment, each elements of color, nearness, and right/left of a cylinder should be bound.



### 3. Pulse neural network

Conventional neural networks code information using firing rate of neurons, but pulse neural networks code information within their pulse firing. Some concrete models are proposed. In our research we adopted the spiking neuron model as neuron model because of its simplicity<sup>5)</sup>

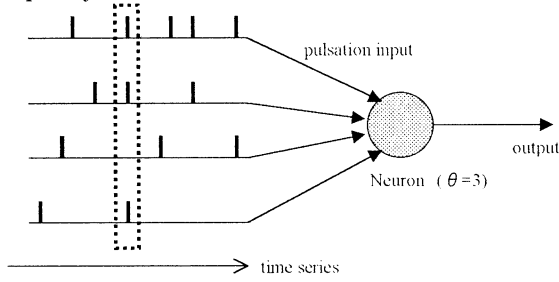


Fig.2 Neuron model of Coincidence detector

Because the threshold of this neuron is 3, only when three pulsation inputs specified by broken lines are inputted, this neuron can fire.

This neuron model function as a coincidence detector that detects coincident arrival of pulsation input as shown in Fig.2. Each neuron has its internal state  $P_v(t)$  defined by a response function  $\mathcal{E}_v(t)$ . The internal state of a neuron  $u$  connected to the neuron  $v$  is defined in the equation (1).

$$P_v(t) = \sum_u \sum_{s \in F_u, s < t} w_{u,v} \cdot \mathcal{E}_{u,v}(t-s) \quad (1)$$

In this equation,  $F_u$  is a set of times of firing for a neuron  $u$ . Response function represents the rate of decay of internal state. If this rate is small the neuron becomes coincidence detector, and if the rate is very large the neuron behaves as an integrator used in a conventional neural network. In our research, we adopt simple function defining only decay constant  $\tau$  as shown in equation (2) for a response function.

$$\mathcal{E}(t) = \begin{cases} 0 \cdots t < 0, \tau < t \\ 1 \cdots 0 \leq t \leq \tau \end{cases} \quad (2)$$

When pulsation inputs are inputted, this coincidence detector neuron raises its internal state for response function and weights of inputs, and if internal state reaches its threshold  $\theta$ , this neuron fires and emits a pulse. The refractory period of this neuron model is realized by changing its threshold for time. In our model, the threshold is a constant and we do not consider refractory period.

In a networked pulse neurons, we specify delay time between neurons that indicates the delay of a pulse from a neuron to the other one connected to it. All synapses have this kind of delay times. We can control firing of neurons by changing the delay times.

The difference between the neurons or the networks of this neuron model, and the conventional rate coding neuron model is summarized in TABLE.1.

	PNN	NN
Information Processing	spatio-temporal	spatial
information	timing of firing	rate of firing
neuron model	coincidence detector	integrator
firing of neuron	asynchronous	synchronous

TABLE 1 difference between Pulse Neural Network and Neural Network

### 4. The solution of binding problem by synchronous firing using pulse neural network

#### 4.1 Temporal coding

Each cylinder has three visual features: color, nearness, and right/left. If there is one object, the features fire simultaneously and are bound to one. So it is possible to recognize the object. But if there are some objects, all features of all objects fire at the same time, so that it is impossible to recognize each object properly.

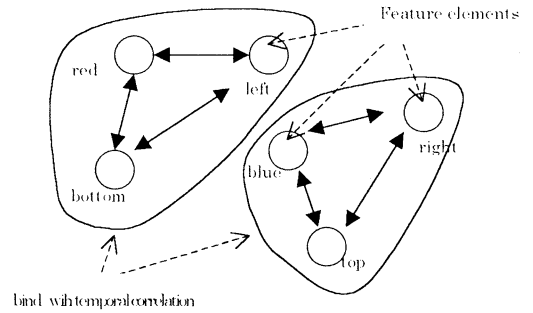


Fig.3. An example of solution of binding problem with temporal correlation

Here, we must take another method to bind the feature. It is possible to recognize each object by binding them using temporal correlation of a neuron or a group of neurons representing visual feature of one object. For example, as shown in Fig.1, by binding the features of an object with temporal correlation (e.g. specific period and so on), we can think of recognizing the object. In this way, if these neurons fire synchronously or if they fire with having temporal correlation, it is possible to distinguish from other objects (Fig.3).

This enables us to represent much information by small number of neurons by interlacing and binding the features with temporal correlation. Moreover, even if the number of neurons in the network is constant, it

is possible to increase the ability of information processing and capacity of memory by extending temporally.

## 4.2 Representation of the Binding Problem with Pulse Trains

Visual information from a camera is converted to a pulse pattern and is inputted to the network. Here the pulse pattern coded information of a red cylinder and a blue cylinder is added and is inputted to the network. In this example, the delay times corresponding to the element of visual features are set below.

color	red	blue
	3	4
nearness	front	rear
	5	7
right/left	left	right
	13	15

TABLE.2 The delay time (ms) corresponding to the elements of visual features

For example, in the condition of Fig.1 where a red cylinder is rear-left and a blue cylinder is front-right, the periodic pulse trains with interval 3,7, and 13(ms) that mean 'red', 'rear' and 'right' and that with interval 4, 5, and 15(ms) that mean 'blue', 'front' and 'left' are combined into a pulse train and is inputted to the network (Fig.4).

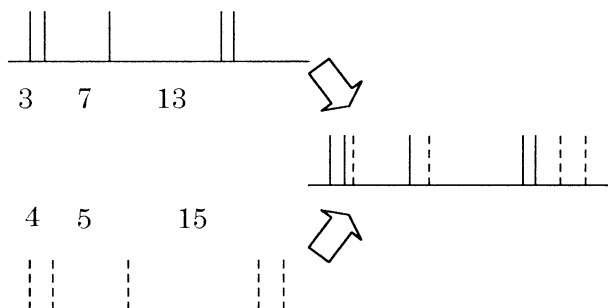


Fig.4 Input pulse train in the case of Fig.1

The connection and delay time between neurons in the network are set for mapping visual inputs and input pulse trains as follows: At first, neurons are allocated to each visual feature. This network consists of six neurons. Each neuron has an input synapse that receives a pulse train from the camera.

The network consists of six neurons corresponding to 'red', 'blue', 'front', 'rear', 'left', and 'right'. A target is indicated by inputting corresponding pulse trains. According to the instruction, the robot identifies the position of a target and invokes a behavior to take it.

With concrete values shown above, we use the network shown in Fig.5.

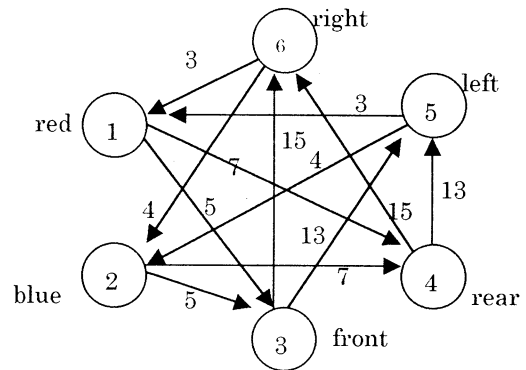


Fig.5 Connection between neurons and their delay times(ms)

These connections are decided according to the guidelines shown below:

1. In order to be able to retain input pulse trains, neurons are connected so as to form closed cycles within visual features.
2. One element of visual feature should be connected to all elements in visual feature connected to it.
3. The delay times of synapses connecting to a neuron should be all the same.
4. We decide these values so that the periods of two pulse trains should not be the same at any conditions.

In this experiment, only one cylinder is a target to be grabbed, but the robot does not know which cylinder is the target. So, robot must understand where two cylinders are placed. Using conventional neural networks, the possible combination of colors and positions of cylinder is  $3 \times 4 = 12$ . Then, in order to be able to represent all conditions, it needs 12 neurons at least. However, the pulse neural network can bind the feature elements of object temporally, so that only a few neurons are necessary. In other words, any conditions that are represented by those six neurons could be bound by temporal correlation. So, it is an advantage that a task can be accomplished by a smaller number of neurons.

## 5 Computer simulation

### 5.1 Experimental setting

We conducted computer simulations based on the settings as described above. In our simulation, we use six neurons in the network.

The threshold of a neuron is 2, initial internal state is 1, the time constant is 500μs, and all the weights of connections are set to 1.

The color of a target is given by inputting inhibitory pulse trains to the neuron corresponding to the

different color. For example, in the case of instructing to grab a red cylinder, inhibitory pulse trains are inputted to the neuron corresponding to blue. This inhibitory pulse train is inputted randomly at a rate of 1 pulse per 2ms.

## 5.2 Results

At first, a red cylinder is placed at the rear-left. In this case, we conducted periodic pulse trains with interval 3, 7, and 13(ms) are inputted. We conducted the same experiment in the case of a blue cylinder on front-right. Fig.6 and 7 show the results of the firing rates at each condition.

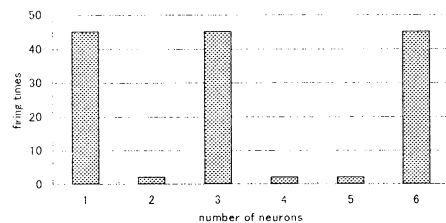


Fig. 6 Firing times of each neuron for (3, 7, 13)

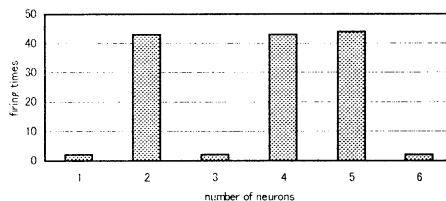


Fig.7 Firing times of each neuron for (4, 5, 15)

Fig.8 shows the case where a red cylinder is on the rear-left and a blue cylinder is on the front-right.

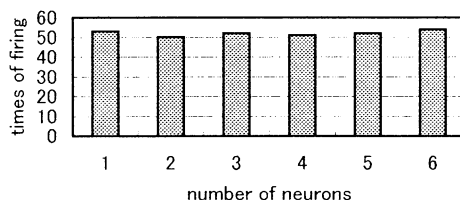


Fig.8 Firing times of each neuron for superposed pulse train

In Fig.8, we can see a slight difference but it can be thought as error.

Then, we instructed to grab either a red cylinder (Fig.9) or a blue cylinder (Fig.10) whereas the positions of two cylinders are the same as before.

From these results, we can conclude that the network for the robot can recognize the positions of two cylinders without confusing and with minimum number of neurons.

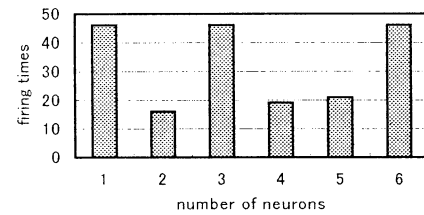


Fig.9 Instructed to grab a red cylinder in the condition of Fig.8.

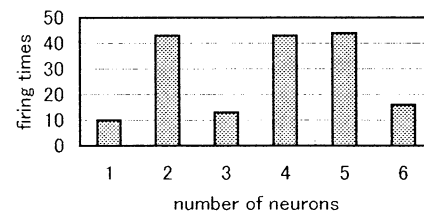


Fig.10 Instructed to grab a blue cylinder in the condition of Fig.8.

## 6. Conclusion

In this paper, we defined the binding problem on making decision of a robot and verified it with computer simulations. We can conclude that using pulse neural networks, the size of network is able to be reduced not causing combinatorial explosion as the conventional rate coding neurons.

At present we must decide delay time of connection by hand. In case of designing much larger networks, we need a kind of learning algorithm according to our guideline shown above, and this is one of our future works.

## Reference

- 1) C. von der Malsburg: Binding in models of perception and brain function, *Current Opinion in Neurobiology*, 5: 520-526, 1995.
- 2) H. Fujii, H. Ito, K. Aihara, N. Ichinose and M. Tsukada: Dynamical Cell Assembly Hypothesis – Theoretical Possibility of Spatio-temporal Coding in the Cortex -, *Neural Networks*, Vol. 9, No. 8, pp. 1303-1350, 1996.
- 3) Natsuhir Ichinose: Dynamical information coding in brain.(in Japanese), *Computer Today*, No. 82, pp. 41-51, 1997.
- 4) C. von der Malsburg: The correlation theory of brain function, *Models of Neural networks II*, Chapter 2, pp. 95-119.
- 5) W. Maass: On the computational complexity of networks of spiking neurons, *Advans in Neural Information Processing System*, Vol. 7, MIT Press, pp. 183-190, 1995.

# RESEARCH ON USING DYNAMIC NEURAL NETWORKS IN MODEL PREDICTIVE CONTROL\*

LI SHU-RONG, LI FENG

Department of Automation, University of Petroleum (East China), Dongying, 257062, P.R.China.  
E-Mail: Lishuron@hdpu.edu.cn

LU QIANG

Department of Electrical Engineering, Tsinghua University, Beijing, 257062, P.R.China.

**Abstract:** An approach of reconstruction a nonlinear dynamics by using dynamic recurrent networks (DRNN) is presented in this paper. The main idea of this paper is to construct an observer-like dynamic recurrent neural network to match a practical process. The network is trained by using input and output information of practical systems. An algorithm for adjusting weight matrix of the network is also given. After a dynamic recurrent neural network is well trained, the so called DRNN based predictive control can be implemented. The well trained dynamic recurrent neural network plays a role as a predictive model in the proposed control scheme. Simulation for a nonisothermal continuous stirred tank reactor (CSTR) system demonstrates the effectiveness of the training algorithm and the control strategy.

**Keywords:** Nonlinear systems, dynamic recurrent neural networks, predictive control..

## 1. INTRODUCTION

Model based predictive control (MPC) has been widely discussed in literatures, Garcia<sup>1</sup>, Clarke<sup>2</sup>, Scattilini<sup>9</sup>. Linear model based MPC can not give satisfactory behaviors when the system has large operation transition and severe external disturbance. Thus, Someone proposed the use of neural network in MPC, Chen<sup>3</sup>, Saint-Donat<sup>4</sup>, Nikolaou<sup>5</sup>, etc.. With the on-line learning ability of neural networks, a well-trained neural network can be used as a predictive model so that the effect of the model/plant mismatch problem can be greatly decreased. Moreover the neural network can predict the future output for the optimal input. According to this idea, a dynamic recurrent neural network (DRNN) based MPC is investigated in this paper. In such a control scheme, the neural network is full recurrent. The training algorithm is also provided.

This paper consists of following main contents: (1). Training a DRNN to match a nonlinear dynamics. An observer-like DRNN is constructed for making use of the information of the coarse mechanistic model of a

practical plant. The advantages of such a DRNN are as follows: (i). save computational time; and (ii). assure the local asymptotic stability of the error dynamics. (2). DRNN based model predictive control. Once a DRNN is well trained, it can be used to produce a dynamic matrix and predict future output. Similar to linear model based MPC the optimal future control strategy in DRNN based MPC can be calculated by on-line minimizing a given performance index. To show the efficiency of the proposed algorithm and controller, simulations for a CSTR system are given in the paper.

## 2. PRELIMINARIES

Consider a following nonlinear control system

$$\begin{cases} \dot{x} = f(x, u) & x \in R^n, u \in R^m \\ y = Cx & y \in R^p \end{cases} \quad (1)$$

where  $x, u, y$  are states, control and output respectively. Assume  $f(0,0) = 0$ .

System (1) can be written as follows

$$\begin{cases} \dot{x} = Ax + Bu + g(x, u) \\ y = Cx \end{cases} \quad (2)$$

$$\text{where } A = \left. \frac{\partial f}{\partial x} \right|_{(0,0)}, B = \left. \frac{\partial f}{\partial u} \right|_{(0,0)}, g = O(\|p\|^{\alpha}),$$

$$\rho = \left( \|x\|^2 + \|u\|^2 \right)^{1/2}, \alpha > 0.$$

It is known that if  $[A, C]$  is observable, then an observer of (1) can be designed with the following form

$$\begin{cases} \dot{\hat{x}} = A\hat{x} + Bu + g(\hat{x}, u) + L(y - C\hat{x}) \\ \hat{y} = C\hat{x} \end{cases} \quad (3)$$

Generally, (3) is a local observer because the error equation is locally asymptotic stable, Qin<sup>8</sup>.

\* This work is partially supported by the NSF of China and the NSF of Shandong province of China.

A dynamic recurrent neural network can be described by a set of nonlinear equations

$$\frac{dx_i}{dt} = -\alpha_i x_i + \beta_i \cdot f_i \left( \sum_j w_{ij} x_j \right) + I_i \quad (4)$$

where  $x_i$  is the state of the  $i$ th neuron,  $w_{ij}$  is the connection strength or weight from  $j$ th neuron to  $i$ th neuron.  $\alpha_i$  and  $\beta_j$  can be chosen as constants, and function  $f_i$  usually have many forms. We select  $f(\xi) = \frac{1-e^{-\xi}}{1+e^{-\xi}}$  in this paper.  $I_i$  is external input.

For DRNN (4), It is known that (i). If the weight matrix  $W$  is symmetric and with zeros in diagonal elements, then DRNN (4) is a Hopfield neural network. (4) is globally asymptotic stable; (ii). If  $W$  is not symmetric, then using the following state transformation  $v_i = \sum_j w_{ij} x_j$ , the DRNN (4) can be

turned to be a Hopfield neural network, and the global stability can be guaranteed, Pineda<sup>7</sup>.

### 3. TRAINING A DRNN TO MATCH A NONLINEAR DYNAMICS

Consider system (1) and assume  $[A, C]$  observable.

Then we can design a matrix  $L$  such that  $(A-LC)$  has negative eigenvalues  $-\lambda_1, \dots, -\lambda_n, \lambda_i > 0$ , and for  $i \neq j$ ,  $\lambda_i \neq \lambda_j$ . Then system (1) can be written as:

$$\dot{x} = (A - LC)x + Bu + g(x, u) + LCx$$

Without loss of generality, we choose a suitable coordinate system, such that  $(A-LC)$  is a diagonal matrix

$$(A - LC) = \text{diag}(-\lambda_1, -\lambda_2, \dots, -\lambda_n), \lambda_i > 0.$$

Constructing a following DRNN: the number of the neurons may be large but in the output layer the number of neurons is constrained to be  $n$ . Using  $\hat{x}_i$ ,  $i = 1, \dots, N$  to represent the states of the neurons where  $N$  is the number of neurons of the DRNN. Then the dynamic equations of neurons in the output layer can be written as

$$\begin{aligned} \dot{\hat{x}}_i &= -\lambda_i \hat{x}_i + I_i + \beta_i \cdot f_i \left( \sum_j w_{ij} \hat{x}_j \right) + \\ &+ \sum_{j=1}^m b_{ij} u + \sum_{j=1}^n (LC)_{ij} x_j, i = 1, \dots, n \end{aligned} \quad (5)$$

where  $(LC)_{ij} = \sum_{h=1}^p l_{ih} c_{hj}$ .

The objective is online training DRNN (5) such that the DRNN accurately matches system (1). Define  $e_i = x_i - \hat{x}_i$ . Then the error equation satisfies:

$$\dot{e}_i = -\lambda_i e_i + g_i(x, u) - (\beta_i \cdot f_i \left( \sum_j w_{ij} \hat{x}_j \right) + I_i). \quad (6)$$

Let the performance index be  $J = \frac{1}{2} \sum_{i=1}^n e_i^2$  or  $J = \frac{1}{2} e^T \cdot e, e^T = (e_1, \dots, e_N)$ . For the neurons not in the output layer, let  $e_i = 0, (n < i \leq N)$ . Write equation (5) in vector form:

$$\begin{aligned} \dot{\hat{x}} &= \Lambda \hat{x} + \beta F(v) + I + Bu + LCx \\ \text{where } \beta &= \text{diag}(\beta_1, \dots, \beta_N), \Lambda = \text{diag}(-\lambda_1, \dots, -\lambda_N), \\ F &= [f_1, \dots, f_N]^T, I = [I_1, \dots, I_N]^T, v = W\hat{x}. \end{aligned} \quad (7)$$

For weight adjusting, the variation of  $W$  is along the negative gradient of  $J$  with respect to  $W$ . i.e.

$$\dot{w}_{pq} = -\eta \frac{\partial J}{\partial w_{pq}}, \quad (8)$$

where  $\eta$  is the training rate.

Refer to LI<sup>6</sup>, the weight is regulated in the following way,

$$\dot{w}_{pq} = \eta \beta_p \frac{\partial f_p}{\partial v_p} \hat{x}_q z_p \quad (9)$$

where

$$\dot{z}_i = -\lambda_i z_i + \sum_k \beta_k \frac{\partial f_k}{\partial v_k} w_{ki} z_k + e_i. \quad (10)$$

### 4. APPLYING DRNN IN MODEL PREDICTIVE CONTROL

For model predictive control (MPC), the future control actions are determined by minimizing the following performance index

$$\min J(k) = \|s_p(k) - \tilde{y}_{PM}(k)\|_Q^2 + \|\Delta u_M(k)\|_R^2 \quad (11)$$

Where  $s_p(k) = [s(k+1), \dots, s(k+P)]^T$  are the  $p$  values of the desired future trajectory;

$$\begin{aligned}\tilde{y}_{PM}(k) &= [\tilde{y}_M(k+1|k), \dots, \tilde{y}_M(k+P|k)]^T \\ &= \tilde{y}_{p0}(k) + A\Delta u_M(k)\end{aligned}\quad (12)$$

is the output prediction model.  $P$  is the prediction horizon,  $M$  is the control horizon,  $Q = \text{diag}(q_1, \dots, q_p)$ ,  $R = \text{diag}(r_1, \dots, r_m)$  are constant weight matrices,  $\tilde{y}_{p0}(k) = [\tilde{y}_0(k+1|k), \dots, \tilde{y}_0(k+P|k)]^T$  are the predicted outputs resulted from former optimal control action.

$$A = \begin{bmatrix} a_1 & & \\ \vdots & & \\ a_M & \dots & a_1 \\ \vdots & \vdots & \vdots \\ a_P & \dots & a_{P-M+1} \end{bmatrix}_{PM}$$

is the dynamic matrix

resulted from the unit step response of the linear model.  $\Delta u_M(k) = [\Delta u(k+1), \dots, \Delta u(k+M-1)]^T$  are the future control actions, Garcia<sup>1</sup>.

From (11) and (12), the optimal future control is

$$\Delta u_M(k) = (A^T Q A + R)^{-1} A^T Q [S_o(k) - \tilde{y}_{p0}(k)] \quad (13)$$

(13) is the well known dynamic matrix control law (DMC). Generally, DMC depends on the linear model of the plant. The linear model based DMC has its shortcoming since in (13) the dynamic matrix is invariant which will result in less satisfactory control behaviors when the controlled plant dynamics is nonlinear, time varying or in presence of severe disturbance. For a nonlinear, time varying system, a little variation of the operation point will lead to a large variation of step response. To overcome the drawbacks in linear model based DMC, a DRNN based DMC is proposed below. A DRNN is trained to match the dynamics of a plant. Then the well-trained DRNN is applied to produce the dynamic matrix and predict future output  $\tilde{y}_{p0}(k)$ . Let  $\hat{A}$  be the dynamic matrix obtained from the well trained DRNN,  $\hat{y}(k)$  be the output of the DRNN, the DRNN based DMC is

$$\Delta u_M(k) = (\hat{A}^T Q \hat{A} + R)^{-1} \hat{A}^T Q [S_o(k) - \tilde{y}_{p0}(k)]. \quad (14)$$

To attenuate the influence of model mismatch, feedback improvement is introduced in (14).

$$\tilde{y}_{p0}(k+1) = \hat{y}_{p0}(k+1) + h \cdot e(k+1)$$

where  $h = [h(k+1), \dots, h(k+P)]^T$  is constant vector which can be pre-assigned.  $e(k+1) = y(k+1) - \tilde{y}(k+1)$  is predictive error in  $K+1^{\text{th}}$  sample time.

To show the efficiency of the proposed DRNN based DMC above, we choose a nonisothermal continuous

stirred tank reactor (CSTR) as controlled plant, Nikolaou<sup>5</sup>. The dynamic equations of the CSTR are described as follows:

$$\begin{aligned}\frac{dc_A}{dt} &= \frac{F_I}{V} [c_{A_I} - c_A(t)] - kc_A(t) \exp\left(-\frac{E}{RT}\right) \\ \frac{dT}{dt} &= \frac{F_I}{V} [T_I - T(t)] - \frac{\Delta H_R}{\rho c_p} kc_A(t) \exp\left(-\frac{E}{RT}\right) - \frac{Q(t)}{\rho c_p V}\end{aligned}$$

Where  $c_A$  is the concentration of species A in the reactor,  $T$  is the temperature,  $Q$  is the heat removal rate (input  $u = (Q - Q_S)/Q_S$ ). Parameter values are shown below.

$F_I$ ( $\text{m}^3/\text{h}$ )	$V$ ( $\text{m}^3$ )	$C_{A_I}$ ( $\text{mol}/\text{m}^3$ )	$k$ ( $1/\text{h}$ )	$E/R$ (K)	$\Delta H_R$ ( $\text{J}/\text{mol}$ )	$T_I$ (K)	$\rho$ ( $\text{kg}/\text{m}^3$ )	$C_p$ ( $\text{J}/\text{kg} \cdot \text{K}$ )
1.133	1.36	8,008	7.08 e7	8,375	-69,7 75	373.3	800.8	3,140

Let  $y = \Delta C_A/C_{AS} = (C_A - C_{AS})/C_{AS}$ . The steady state is  $(Q_s, c_{AS}, T_s) = (1.055 \times 10^8 \text{ J/h}, 393.3 \text{ mol}/\text{m}^3, 547.556 \text{ K})$ .

A 7-node DRNN is trained to match the dynamics of the CSTR. The input and output data for training the DRNN are generated by adding 50% pseudo-random number sequence (PRNS) signals to the steady-state input of the plant. The sampling period is 0.07h and 1000 pairs of input and output data are obtained in Fig.1 and Fig2. After training, different PRNS input signals are used to test the matching ability of the DRNN model in Fig. 3. Fig.4 is the comparison of the step response of the CSTR with that of DRNN. Once the DRNN is well trained, a dynamic matrix can be achieved by adding a step signal to this DRNN which is used as a predictive model. The DRNN based DMC is (14). To illustrate the property of the DRNN based DMC, a 25% step signal is added to the closed-loop system initial set-point. A 7% disturbance is added to steady output of the controlled system to test the ability of disturbance rejection which is shown in Fig.5.

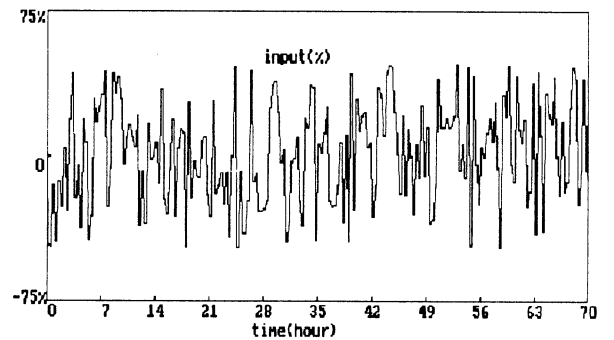


Fig.1. Training input data for DRNN

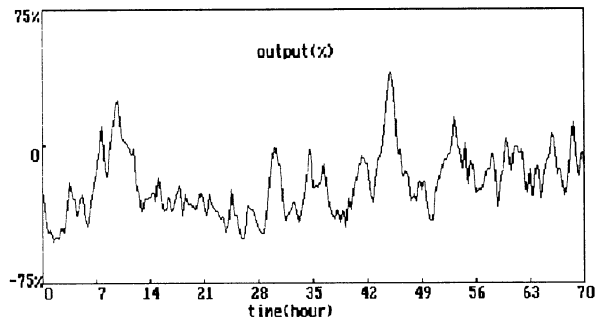


Fig.2. Training output data for DRNN.

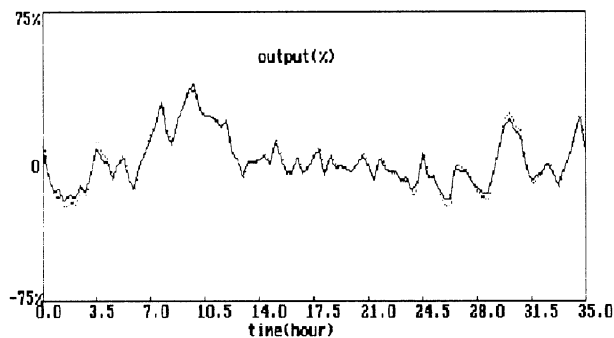


Fig.3. CSTR output (—) and DRNN output(...) Corresponding to the test input

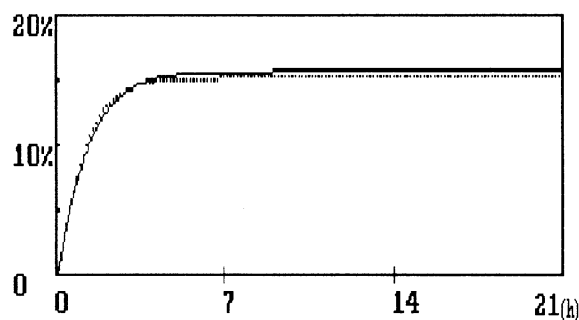


Fig.4. Step responses of CSTR(—) and DRNN(...).

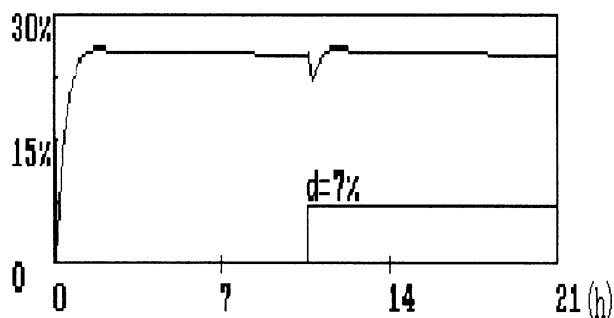


Fig.5. Disturbance rejection by using DRNN based MPC.

## 5. CONCLUSIONS

Two problems are treated in this paper. One problem is the training of a DRNN to match a nonlinear dynamics. The DRNN used in this paper is full recurrent and with an observer-like form. A learning algorithm is provided. The advantage of the DRNN is that it can learn not only the static relation but also the dynamic relation. The other problem is the discussion of the role of a DRNN played in MPC. A well-trained DRNN can provide a dynamic matrix on-line as well as the prediction of the future outputs. Compared with linear model based MPC, DRNN based MPC can provide a more accurate predictive model since the DRNN can be trained periodically so that it can overcome the effects caused by time-varying, non-linearity and external disturbance existed in the practical systems. The limitation of the DRNN based MPC is that it takes long time to train a DRNN since an auxiliary system is involved in the learning algorithm. In fact, to reduce computation time, we can use partial recurrent neural network instead of full recurrent one. The DRNN based MPC can widely be applied in process control.

## REFERENCES

- [1] Garcia C.E, Prett D.M., Morari M.(1989) Model Predictive Control: Theory and Practice—a Survey, *Automatica*, 25(3), 335-348
- [2] Clarke, D.W., Mohtadi, C., Tuffs, P.S., (1987) General predictive control, *Automatica*, Vol.23, No.3, p137-160.
- [3] Chen, Q., W.A. Weigand (1994), Dynamic optimization of nonlinear processes by combining neural net model with UDMC. *AIChE Journal* 9:1489-1497
- [4] J. Saint-Donat, N. Bhat, T.J. Mcavoy (1991), Neural net based model predictive control. *Int. J. control* 6:1453-1468.
- [5] Nikolaou, M., V. Hanagandi (1993), Control of nonlinear dynamical systems modeled by recurrent neural networks. *AIChE Journal* 11:1890-1894
- [6] LI, S.R., Li, F., (1998) Reconstruction of a nonlinear dynamics via a dynamic neural network, in *proceedings of the Int. Con. On systems science*, Poland, Volume II, 287-293.
- [7] Pineda, F.J., (1987) Generalization of back-propagation to Recurrent neural networks, *Physical Review Letters*. Vol 59. 2229-2232.
- [8] Qin, H.S., LI, S.R., Dynamic compensation for a class of nonlinear control systems, *Control & Decision*, (in Chinese) Vol.4, No. 4, 1989.
- [9] Scattolini, R., Schiavoni, N. (1995) A Multi-rate model based predictive controller, *IEEE Trans. Auto. Control*, 40, 1093-1097.
- [10] Williams, R.J., D. Zipser, (1989) A learning algorithm for continually running fully recurrent neural network, *Neural Computation*, 1, 270.

## Adaptive Positioning of Soccer Agents with Hybrid Learning System

Naoyuki AKIYAMA

Keiji SUZUKI

Masahito YAMAMOTO

Azuma OHUCHI

Laboratory of Harmonious Systems Engineering,  
Research Group of Complex Systems Engineering,  
Graduate School of Engineering, Hokkaido University  
Kita 13, Nishi 8, Kita-ku, Sapporo, 060-8628 JAPAN

### Abstract

In this paper, we propose the method of adaptive positioning of the soccer agents on the soccer simulator field for realizing the cooperative pass play. The goal of this "cooperative pass play" is that one player feeds a ball to an appropriate position, where a defender can not intercept the ball, and the other player moves to the position in order to receive the pass.

In addition, we adopt the hybrid learning system - combined neural network, learning classifier system, and Q-learning - proposed by Robert E. Smith [1] in order that soccer agents may learn autonomously to decide appropriate position in accordance with the environmental states to complete pass play.

By using the agent implemented the method, we try to realize the cooperative pass play in the two-on-one situation on the RoboCup Soccer Server[2].

### 1 Introduction

In real soccer games, ball possession time for single player is very short. So each player has to move without a ball (generally called *free running*) most of a game. About pass play, receiver's free running prevents fixation of pass causes, and varied pass courses may decrease the probability of pass interception by a defender. For this reason, we think free running is one of very important factors to complete pass play, so we intend to realize cooperative pass play by the agents doing free running on the soccer simulator field.

To realize it, we arrange free running as "movement to an appropriate position on the field in accordance with environmental information". We call this arrangement method *adaptive positioning*.

### 2 Architecture of Soccer Agent

To implement the method of adaptive positioning, we construct the action-execution architecture of the agents. We design the agent's basic architecture based on "ogalets", the winner of Pre-RoboCup'96[3]. On ogalets, the agent can select several actions (pass, shoot, search a ball, approach to a ball, and go back to a own home position) around fixed home position. The selection of the actions is done in accordance with the distances between own home position, a ball, and other agents.

In order to adopt the proposed method, we divide the half field to the 5×6 square "zones", and we define the center of each area as the "position". After that, we arrange this architecture as the agent can move all around the field by position change. The basic action algorithm of the agent is as follows.

1. Receive environmental information
2. Decide which position to move
3. Move to the position
4. Search a ball and judge ball existing zone
  - If ball on agent existing zone, agent can kick.
  - Ball on other zone, agent only watch the ball.
5. Kick or Watch the ball

The focus of this architecture is the decision making of "where to move" and "where to kick". For these decision making in accordance with environmental states, the agent should have the table of the fitness values of all positions.

We intend that the agent can learn to create this table autonomously. So we adopt the hybrid learning system[1] as the upper layer of the architecture for decision making function.



### 3 Hybrid Learning System for Soccer Agent

This learning system is the integration of neural network, learning classifier system, and Q-learning. This system is based on a three-layered neural network. Each hidden layer node of this network is represented by a classifier. And the network is used to map environmental states to Q values.

In our agent's architecture, Q values mapped by the network mean the fitness values of each position associated with an environmental state. By referring the Q values from the system, our agent can select the appropriate position associated with received environmental state.

#### 3.1 The Network of This System

The input of this system is the current environmental state. And with relation to hidden layer, environmental states should be represented by binary like form. In this system, the state is formed by the bipolar representation. In the bipolar representation, 0 in the binary is represented by -1, 1 is represented by +1, and # (don't care) is represented by 0.

On the Soccer Server, an environmental state should mean the location (absolute or relative) of each object (ball, player, goal) on the field. And in our adaptive positioning method, the agent needs to judge that in which zone each object exists.

Therefore, we transformed the absolute location of an object to bipolar representation of its existing zone. Number of zone is  $6 \times 5 = 30$  as mentioned in section 2, length needed for representation of single object is 5.

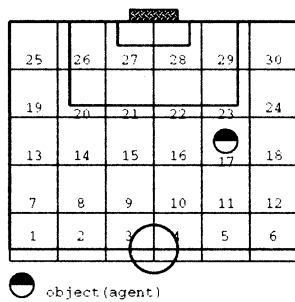


Figure 1: An example of location

For example, an agent exists in zone 17 in Figure 1. 17 can be transformed to 10001 in 5-length binary representation. This is transformed to 1-1-1-1 1 in bipolar representation.

In this network, each hidden layer node represents a classifier. These nodes have a threshold activation function, and compete for the opportunity to fire. The matching between the input and each classifier node is the procedure for exceeding a threshold. Moreover, each classifier node has Q values assigned to all possible actions. The values are used for the weights of the connections to the output layer from the hidden layer.

Each output node corresponds to a possible action. All output nodes sum the Q values from the hidden layer, then the node which has the max Q value is allowed to output.

In our method, possible action means moving to a position or kicking the ball to a position. So the total possible action is  $30 \times 2$ , 30 is equal to a number of positions. So we gave each classifier node 2 tables of Q values, one for moving position selection, the other for kicking position selection.

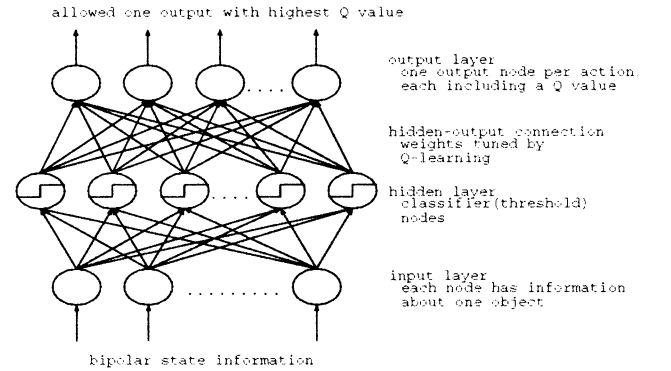


Figure 2: A hybrid learning system

#### 3.2 Evaluation and Genetic Operation

After doing selected action by the output, the Q values in the classifier nodes associated with output are updated in the way of Q-learning. Update of common Q-learning is as follows:

$$Q_{t+1}(i, u_t) = Q_t(i, u_t)(1 - \alpha) + \alpha[r_i(u_t) + \gamma \max_j Q_t(j, u_{t+1})] \quad (1)$$

where  $0 \leq \alpha \leq 1$  is a *learning rate*,  $0 \leq \gamma \leq 1$  is a *discount parameter*,  $Q_{t+1}(i, u_t)$  is a Q value assigned to action  $u$  in state  $i$  at time  $t$ ,  $r_i(u_t)$  is the reward at  $t$ , and  $\max_j Q_t(j, u_{t+1})$  is the maximum Q available in state  $j$ . State drifts from  $i$  to  $j$ . However, because of the dynamics and complexity of soccer field (mainly because influence to the field by other players), it is

impossible to get  $\max Q_t(j, u_{t+1})$  at any simulation step. So we delete  $\max Q_t(j, u_{t+1})$  from (1). Update is as follows:

$$Q_{t+1}(i, u_t) = Q_t(i, u_t)(1 - \alpha) + \alpha \cdot r_t(u_t) \quad (2)$$

After updating Q values for a period, the GA operations are executed to improve classifier nodes.

**Selection** is the roulette wheel parent selection based on the strength of each classifier node. The strength is determined by summation of all Q values.

**Crossover** is the one-point crossover on the condition part of classifier nodes, and on the sequence of Q values that the node has.

**Mutation** is done only to the condition part of classifier nodes. 1 to -1, -1 to 1, 0 to 1/-1.

## 4 Experiment

To confirm the performance of our agent with proposed method, we have set a simple two-on-one situation on the Soccer Server, and made an experiment to realize a cooperative pass play by our agents in the situation.

### 4.1 Situation for experiment

Two kinds of players play in the situation. One is **attackers**. Attackers consist of 2 players, the purpose is to get goal with cooperative pass play. The other is **defender**. Defender consists of 1 player, the purpose is to prevent attackers by intercepting their pass. Attackers are performed by our agent. Defender is an simple agent that only pursues a ball.

Initial location of attacker1 is on zone4, attacker2 on zone2, defender on zone16, and ball on zone4. To prevent the goal by an accidental long shoot, attackers restrict their shoot zone. attackers can only shoot in **shoot area** (See Figure3).

Each experimental trial begins when attacker1 kicks the ball. A trial finishes when attackers get a goal(**complete**), or a defender intercepts their pass(**incomplete**). When a trial finishes, each player and ball moves to initial location, and new trial starts.

### 4.2 Attackers detail

In this situation, informations of objects that can be received are absolute position of a ball, self, another attacker(=teammate), and a defender. As described in section3.1, information of one object is encoded 5-length bipolar, the length needed for an input is  $5 \times$

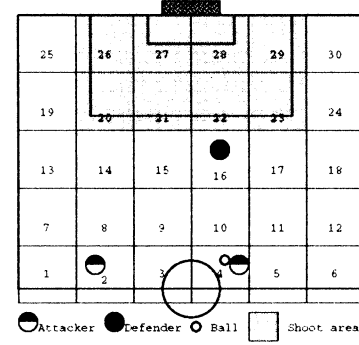


Figure 3: Experimental situation

$4 = 20$ . In addition to the length, we added 2 bid for representing vector of ball. Therefore, the total length of an input is  $(5 + 2) + 5 \times 3 = 22$ .

On the other hand, in this situation attacker1 kicks the ball to an appropriate position at first, then moves to the position in shoot area and shoot if possible. Attacker2 moves to an appropriate position at first, then kick the ball to the position in shoot area and shoot if possible. For these reason, Attacker's total possible action is 38. In attacker1, 30 for kicking position selection, 8 for moving position selection. In attacker2, 30 for moving position selection, 8 for kicking position selection. We call set of 30 Q value in each attacker  $Q1$ , and set of 8 Q value  $Q2$ .

Every kind of parameters for the learning system as follows.

length of classifier nodes	20
size of $Q1$	30
size of $Q2$	8
initial value of each Q value( $INIT\_Q$ )	100
leaning rate( $\alpha$ )	0.3
crossover rate	0.6
mutation rate	0.15

And reward is as follows. Reward to  $Q1$  is  $R1$ , to  $Q2$  is  $R2$ . Reward to attacker1 is  $R_a$ , to attacker2 is  $R_b$ .

- Incomplete pass from attacker1 to attacker 2  
 $R_{a1} = Q1 - Q1/10 \cdot (300/e\_time)$   
 $R_{b1} = Q1 - Q1/10 \cdot (m\_dist/87) + 87/g\_dist$   
 $R_{a2} = R_{b2} = Q2$
- Incomplete pass from attacker2 to attacker 1  
 $R_{a1} = Q1 - Q1/10 \cdot (300/e\_time)$   
 $R_{a2} = Q2$   
 $R_{b1} = Q1 + INIT\_Q \cdot (e\_time/300)$   
 $R_{b2} = Q2 - INIT\_Q/10 \cdot (300/e\_time)$

- Get Goal

$$R_{a1} = R_{b1} = Q1 + INIT\_Q \cdot (300/e\_time)$$

$$R_{a2} = R_{b2} = Q2 + INIT\_Q \cdot (300/e\_time)$$

$e\_time$  : the time by pass incompletion

$m\_dist$  : the movement distance of the agent

$g\_dist$  : the distance from the agent to goal

We define the “period” as time for getting 2 goals by attackers. Attackers execute GA operations to their classifier nodes at the end of a period. We measure the time of a period continuously 300 times. The time is expected to be the guideline of the performance of our agent.

## 5 Result and Discussion

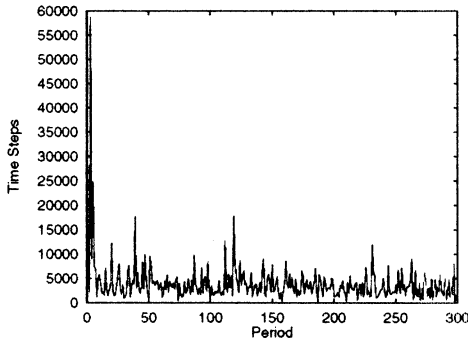


Figure 4: Times for a period

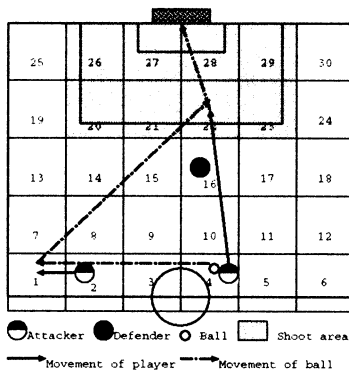


Figure 5: Generated play

Figure4 shows the transition of the time of a period. At early period, the time is very large. But as progres-

sion of the period, the time decreases rapidly and after period 150, the time converges at about 3000~5000.

Figure5 shows the generated cooperative pass play at the end of the experiment. Attacker1 passes to zone1. After kicking, attacker1 moves to zone22 and waits a ball. At the same time, attacker2 moves to zone1 and waits a ball. When a ball comes to zone1, attacker2 aims to pass to zone29. In zone22, attacker1 shoots the ball that is on the way to zone29.

From figure4 and 5, it is to say that the learning system can be effective for reducing search space of position selection. and that the agent can generate a cooperative pass play with proposed method. This method is expected to be effective at more complex situations.

But the agent has a problem. As figure4 indicated, the converged time of a period is not already small. In short, pass incompletion often occurs at a later time. Most of pass incompletion is the pass from zone1 to zone29 by attacker2. We think that the reward without consideration for difficulty of kick occurs the problem. The definition of rewards should be well considerable.

## 6 Conclusion

We proposed the method of adaptive positioning for an accomplishment of cooperative pass play. And to realize it, we constructed the agent with hybrid learning system. As a result, the agent could realize the wall pass like play in the two-on-one situation. From the result, it is to say that the method is effective for the pass play to a certain extent.

## References

- [1] R.E.Smith and H.B.Cribbs III, “Combined biological paradigms:A neural, genetics-based autonomous system strategy” *Robotics and Autonomous Systems*, Vol.22, No.1, pp. 65-74, 1997
- [2] RoboCup homepage,  
<http://www.Robocup.org/02.html>
- [3] Kouichi Ogawara’s Program, “Ogalets6”  
<http://ci.etl.ho.jp/noda/soccer/client/index.html#PreRoboCup96>
- [4] Sean Luke, “Genetic Programming Produced Competitive Soccer Softbot Teams for RoboCup97” *Genetic Programming*, pp214-222, 1998

# Recent Design of Motion Planner and Behaviors for Soccer-Playing Robots

Sun-Gi Hong, Tae-Dok Eom, Choon-Young Lee, Min-Soeng Kim, and Ju-Jang Lee

Department of Electrical Engineering

Korea Advanced Institute of Science and Technology

373 - 1 Kusong-dong, Yusong-gu, Taejon 305 - 701, KOREA

TEL : +82-42-869-3432, E-mail : jjlee@ee.kaist.ac.kr

## Abstract

*One of the most important considerations in the robot soccer game is to develop the well-behaved robots. If one can place the robot on the desired position with the desired orientation freely, it is very easy to develop the strategy and to win the game. In this paper, we will focus on how to design the motion planner of robots and how to obtain their behaviors using the proposed motion planner. In the proposed motion planner, we generate the curvature-based path and compute the proper commands for robots. Behaviors such as kicking, returning, and so on can be easily configured by the synthesis of the curvature-based paths. Through some experiments, the efficiency and validity will be shown in this paper.*

## 1 Introduction

A robot soccer game has been attracted many of researchers interest in the field of multi-agent system and artificial intelligence due that it can be a test-bench for various kinds of algorithms. [1, 2] The final and universal objective in robot soccer game, one of the most popular applications of multi-agent system, is to maximize the performance of the robot system by cooperation given some tasks under the robot's physical limitations and the adverse of dynamic environment caused by other agents or objects.

A key factor in a good performance depends mainly

on if each robot shows good basic behavior, path following. Therefore, we propose a curvature-based motion planner, which generate a path considering present and final robot position and orientation. Most strategic behaviors are made by the combination of the curvature-based path, which robot can track easily when host computer transmits command signals calculated from the curvature of the path. The main advantage of the proposed curvature-based path planning is the smooth robot motion and the guaranteed behavior with small tracking error when robot dynamics are identified in advance. Now we will discuss about the proposed method in the following sections.

## 2 Motion Planner for Soccer-Playing Robots

### 2.1 Preliminaries

It is well known that the differential drive mechanism belongs to the nonholonomic system so that it is very difficult to design the motion controller [4]. To lessen the effort of the design, we propose the method of generating the curvature-based path and tracking it in this paper.

In general, the kinematic model of the wheeled mobile robot is given as follows.

$$\dot{x} = v \cos \theta$$

$$\dot{y} = v \sin \theta$$

$$\dot{\theta} = w \quad (1)$$

where  $v$  is a linear velocity and  $w$  is an angular velocity. The values of  $x$ ,  $y$ ,  $\theta$  are all measured with respect to the world coordinate. This model takes into account the nonholonomic constraint

$$\dot{y} \cos \theta - \dot{x} \sin \theta = 0 \quad (2)$$

that specifies the tangent direction along any feasible path for the vehicle. In the above equation,  $v$  and  $w$  can be described as follows.

$$\begin{aligned} v &= \frac{v_L + v_R}{2} \\ w &= \frac{v_L - v_R}{2l} \end{aligned} \quad (3)$$

where  $v_L$  and  $v_R$  are the left and right wheel's velocity respectively and  $l$  is the size of the wheel base. Now the curvature is defined as follows.

$$\begin{aligned} k &= \frac{w}{v} \\ &= \frac{v_L - v_R}{l(v_L + v_R)} \end{aligned} \quad (4)$$

where the value of  $k$  represents the degree to which the path is curved, for example, the straight path has the zero curvature because it is not curved. If the curvature of the given path is known at all the points on the path exactly, then the exact velocity commands can be computed by using the equation (2-4). The basic idea of this paper is that after generating the curvature-based path which can implement the desired motion, we compute the desired velocity command of each wheel. To design the tracking controller for velocity we need to take into account the dynamics of the robot such as the mass distribution, the inertia, and so on. However it is very hard to obtain the dynamic model of a real robot so that we decide to limit the maximum value of the desired velocity after finding out the allowable region where the given velocity to the real robot satisfies the equation (2-4).

## 2.2 Validation experiment

In order to verify the allowable maximum value of the desired velocity, we perform some validation experiments. Now the equation (2-4) can be rewritten

as follows.

$$k = \frac{1 - \alpha}{l(1 + \alpha)} \quad (5)$$

where the value of  $\alpha$  represents the ratio of each wheel's velocity, that is,  $v_R/v_L$ . While keeping the value of  $\alpha$  constant, we increase the left or right wheel's velocity and observe the resulted path. In the allowable region, we can expect that the result paths from another velocities will be the same. As the given velocity grows up, the resulted path will disobey our expectation, then we can find out the allowable region. Figure 1 shows how to perform the validation experiment and check the allowable velocity range. During the experiment, we experienced that the slippage is the major reason of dissatisfying the equation (2-4) so that the careful design of wheels is needed to develop a real robot.

## 2.3 Generation of curvature-based path

In this section, we explain why we need the curvature-based path and how we generate it. The objective of motion planning is to generate the shortest continuous path can place the robot on the desired position with the desired orientation at the end of the path.

In Fig. 2, a robot is assumed to be positioned on point  $R$ . According to the orientation of the robot, we divide the orientation range into three regions. A basic idea is that if a robot is located on point  $C$  and it directs to point  $D$ , then an exact curvature can be computed because there is a circular path from point  $D$  to  $O$ . The current orientation of the robot determines the desired position for the robot to travel, that is, the point  $A$ ,  $B$ , or  $C$ . After iterating the moving motion, the robot eventually reaches the point  $C$  where an exact curvature can be computed then we can obtain the circular path that satisfies the final orientation constraint. In this motion, the point  $C$  is dynamically moving. This motion planning is very simple and powerful in real experiments.

### 3 Synthesis of behaviors

#### 3.1 Kicking Motion

In this behavior, the final desired position will be the ball position and the final desired orientation should be aligned with the attacking direction. After setting these values, it is very simple to implement the kicking motion because we just generate the curvature-based path and compute the desired velocity commands that enable the robot to track it. However, in the real game, some heuristics are needed to determine the attacking direction and position because there is an opponent goalie. By adjusting the attacking direction, the turning kick can be often performed. However, its motion is not robust so that we need to enhance it in the future contest.

#### 3.2 Returning motion

When the kicker robot fails to shoot the ball to the opponent goal, it is essential for returning it to the defensive area in order to block the offense. The major consideration of this motion is to prevent the robots from dribbling the ball into our goal area. In the previous work [3], we proposed the virtual target algorithm that assigns the virtual attraction point behind the ball position and enables the robot to avoid the ball while tracking the circular or arc path. Using the similar idea, we generate the shortest circular or arc path in order to block the ball as shown in Fig. 3. Now we have two basic motions and need to the elaborated strategy using these motions.

#### 3.3 Goalie's strategy

A goalie has a quite simple strategy as follows: it stays between the ball and the goal line, so that the back and forth motion is quite enough to block the ball with its side. The simple motion does not consider the goalie's kicking motion because the probability of losing the points can be increased during the kicking motion. The goalie's motion is limited so that the goalie always stays between goal posts.

#### 3.4 Field players' strategy

The objective of field player robots is as follows: the field player robot kicks the ball into the opponent's goal and when such a kick is not possible or the opponent's attack is aggressive, it will try to improve its position in the playing ground. To achieve this objective, it is very important to know what is going on the playing ground. The starting point of determining kicking, placing, or returning is to observe the relative position between the robot and the ball. If the robot places behind the ball, that is, the ball directs toward the opponent goal area, then it means that kicking is possible. Otherwise, the field player should be returned to block the opponents' attack. Also, the field player sometimes needs to be placed at some position in order to coordinate with the other field player. To develop more elaborated criterion for determining these motions, we divide the playground into grids and develop some rules of selecting motions. Figure 4 depicts one example of determining the motion.

### 4 Concluding remarks and further works

In this paper, we proposed the motion planning method based on the curvature-based path. The proposed method have some advantages in providing an easy way to generate the desired velocity of the robot and satisfying the constraint such as the final position and orientation at the end of the path. After performing the validation experiment, we obtained the maximum allowable value of the desired velocity at first. In the allowable velocity range, the dynamics of the robot has no effect on tracking the desired velocity so that the consideration of the dynamics is not needed any more. The curvature-based path was easily organized by synthesizing some arcs and straight paths according to the relative geometry between the robot and the final position including their orientations. The synthesis of behaviors such as kicking and returning was done by simply assigning the end of the path with the desired orientation. These motions were

combined with the rule-based strategy and show the effectiveness in a real robot soccer game. One of major contributions of this paper is to lessen the effort of designing the motion controller for nonholonomic system without any complex procedures. In the further work, we will develop more elaborated robot and strategy and alleviate large circular motion which became the defects of the proposed method in some situations.

## References

- [1] J. -H. Kim, H. -S. Shim, H. -S. Kim, M. -J. Jung, I. -H. Choi, and J. -O. Kim, "A Cooperative Multi-Agent System and Its Real Time Application to Robot Soccer," *Proc. of IEEE Conf. on Robotics and Automation*, pp. 638-643, Albuquerque, New Mexico, Apr., 1997.
- [2] *Proceedings of Micro-Robot World Cup Soccer Tournament*, Taejon, Korea, Nov., 1996.
- [3] *Proceedings of Micro-Robot World Cup Soccer Tournament*, Taejon, Korea, 1997.
- [4] Jean-Paul Laumond, "Controllability of a Multi-body Mobile Robot," *IEEE Trans. on Robotics and Automation*, Vol. 9, No. 6, pp. 755-763, Dec., 1993.

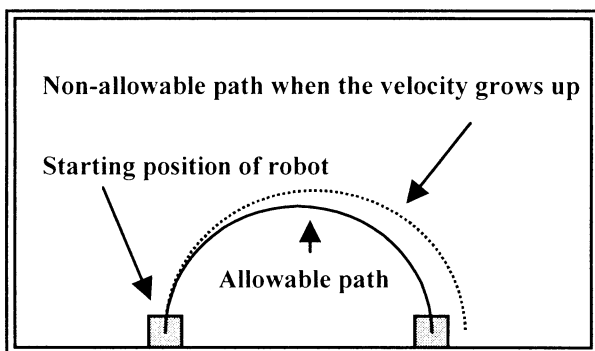


Figure 1: Comprehensive illustration of the validation experiment

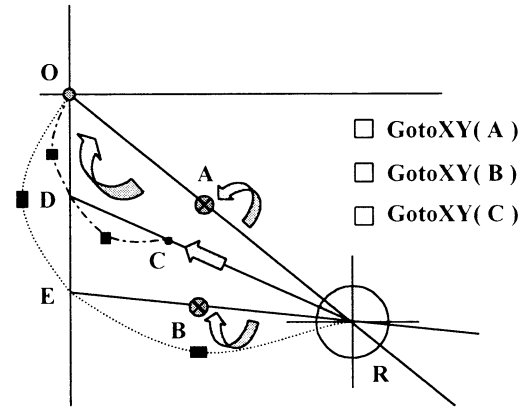


Figure 2: Geometric representation of how to generate a curvature-based path

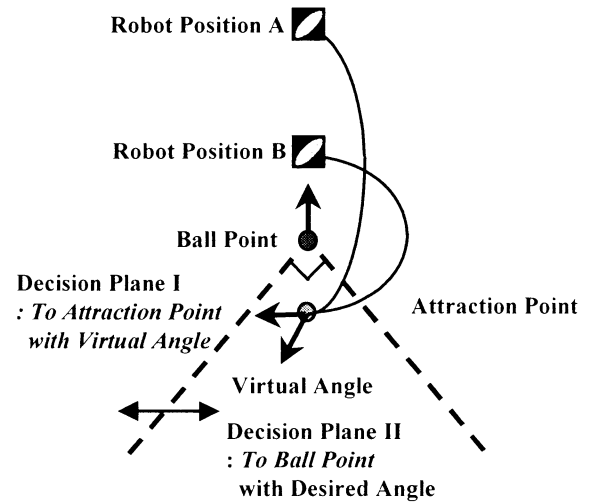


Figure 3: Returning motion

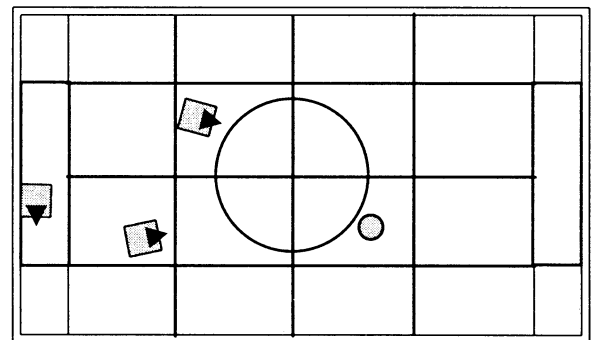


Figure 4: Rule-based strategy

# Neural Network Architecture Optimization and Application Using Genetic Algorithm

Zhi-Jun Liu and Masanori Sugisaka

Department of Electrical and Electronic Engineering  
Faculty of Engineering  
Oita University

Tel: +81-975-54-7831, Fax: +81-975-54-7841  
zjliu@cc.oita-u.ac.jp msugi@cc.oita-u.ac.jp,

**Keywords:** Artificial Neural Networks, Architecture Optimization, Genetic Algorithm, Back Propagation Algorithm, Nonlinear Function.

## ABSTRACT

In this paper, we consider the problem of neural network architecture optimization. Genetic algorithm (GA) is implemented to search for the optimal structures (i.e. the kind of neural networks, the number of inputs and hidden neurons) of neural networks which are used approximating a given nonlinear function. Two kinds of neural networks, i.e. the multilayer feedforward [1] [2] and time delay neural networks (TDNN) are involved in this paper. The weights of each neural network in each generation are obtained by back propagation (BP) algorithm. The simulation results are given out and some improvements in the future are outlined.

## 1. INTRODUCTION

Artificial neural networks have been successfully implemented in various application fields in the last decade, e.g. function approximation [3], classification and time series prediction. Most of the neural networks applied are kind of ones with fixed architectures, it means that we do not know exactly what kind of neural network, the related training algorithm and how many inputs and the hidden neurons will be mostly suitable to the special object. We have some simulation results [4], [5], [6] on the function approximation application based on the neural network, in which, we tried one by one manually to search for the most suitable neural network structure for an multi-input one output nonlinear processing in a temperature control system. Finding an alternative to automatically optimize the neural network architecture is our recent research emphasis.

To design the optimal architectures of artificial neural networks (ANNs), one attempt is implementing the evolutionary algorithms, e.g. evolutionary programming (EP) [7], genetic algorithm (GA). There are also various designing methods relating to the automatically searching for the optimal architectures of ANNs, e.g. constructive and pruning algorithms [8].

In this paper, GA based algorithm is proposed to

search for the optimal architecture of ANNs which are used to approximate a given nonlinear function.

The initial population is generated randomly with some constrains, then the initial generated neural networks are trained by some kinds training algorithms (Back Propagation is selected at first) to sort according to the fitness of each one. The genetic operators are applied to the chromosomes in the same generation and then the next generation is generated. The GA processing stops till one of certain stop criterias (e.g. the defined elapse time, the defined generations or the output accuracy of neural network) is reached. Some of parameters and coefficients are adjusted to see the affects of the neural network output.

The simulation results show that this kind of optimal architecture of neural network is much suitable to the real system than that generated manually by experience.

## 2. GA BASED OPTIMAL ALGORITHM

### 2.1. Encoding of Neural Networks

The first step in GA processing is encoding the neural networks into binary strings called chromosome. A chromosome's characteristic is determined by the genes which are presented by binary bits in this paper. There are numerous encoding algorithms in neural network optimization. Because we emphasis our research on the neural network architecture optimization, the encoded strings just contain the architecture information of ANNs, i.e. the weights of ANNs will not be involved in the encoded strings. The job of obtaining suitable weights is fulfilled by using the ANN training algorithms.

The general description of encoding method is shown by Fig. 1.

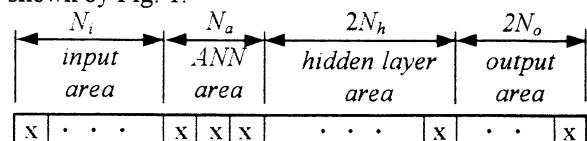


Fig. 1 The chromosome presentation



Fig. 1 shows one chromosome related to one neural network architecture, where  $x$  is one binary bit representing '0' or '1'. The chromosome consists of four parts, i.e. input area, ANN area, hidden layer area and output area, respectively. The length of each parts is  $N_i$ ,  $N_a$ ,  $2N_h$  and  $2N_o$  bits, respectively, where  $N_i$ ,  $N_h$  and  $N_o$  is the number of inputs, hidden neurons and outputs of ANN, respectively.

In this paper, we just deal with the architecture optimization of multilayer feedforward full connected neural network. In input area, one bit represents one input in input layer of neural network, with '1' representing the existence of the input and '0' representing the absence of the associated input. In the second portion of the string, three bits are assigned to represent different kinds of ANNs, with '001' representing multilayer feedforward neural networks, '010' representing TDNN, etc. In hidden layer and output areas, two bits represent one neuron because we consider three kinds of active functions for each neuron, with '00' represents the absence of the neuron, '01' --- the neuron with linear function, '10' --- the neuron with sigmoid function, '11' --- the neuron with hyperbolic tangent function, respectively.

Fig. 2 shows one of multilayer feedforward neural networks generated by GA randomly, with the capacity of 4 inputs, 5 neurons in the first and second hidden layer, one neuron in output layer.

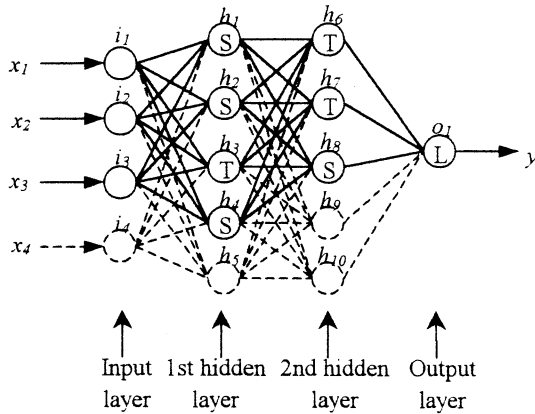


Fig. 2 One of multilayer feedforward neural networks generated by GA

In Fig. 2 the dotted lines imply the absence of the associated input joints or the neurons and the related connections. The real lines imply the existence of neurons and connections. Using the encoding technique illustrated in Fig. 1, the related chromosome of neural network is shown by Fig. 3.

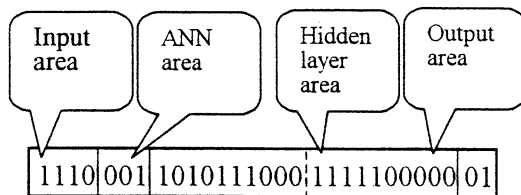


Fig. 3 The chromosome of neural network

## 2.2. The Initial Population and Fitness Function

The initial population is generated randomly, then each chromosome in the initial population is decoded to the associated neural network architecture. Giving the initial weights of the network randomly, the networks are trained by using the general back-propagation algorithm. The fitness of each fully trained network is calculated using equation 1.

$$J_f = \frac{1}{\delta} \left( 1 + 0.5 \left( \alpha \left( 1 - \frac{N_i}{N_{i_{max}}} \right) + \beta \left( 1 - \frac{N_h}{N_{h_{max}}} \right) \right) \right) \quad (1)$$

where  $J_f$  is the fitness of the neural network in one population,  $\delta$  is the error of ANN,  $\alpha$  and  $\beta$  are coefficients implying the influence of inputs number and hidden neuron number,  $N_{i_{max}}$  and  $N_{h_{max}}$  is the maximum number of inputs and hidden neurons, respectively. In this paper, several selection of  $\delta$  are available to calculate the fitness of each ANN, e.g. average absolute error (AAE), root mean square error (RMS) and mean square error (MSE).

All of the networks in one generation are sorted by their fitness after full training. To guarantee each ANN in one generation be fully trained, two coefficients, i.e.  $k_s$  and  $k_e$  are implemented to start and end the fitness calculation. The fitness will not be calculated and the associated ANN will not be sorted before  $k_s$  presentations, and to stop the training of this ANN after  $k_e$  presentations if the error does not decrease. GA operators, i.e. reproduction, crossover and mutation are implemented one generation by generation to search for the optimal architecture of neural network.

## 2.3. Reproduction

One commonly used technique is roulette wheel reproduction. It can be regarded as allocating pie-shaped slices on a roulette wheel to population members, with each slice proportional to the member's fitness. There are two drawbacks to this method: it is possible that some of the best individuals may not be reproduced at all, and thus their genes may be lost. Also, it is possible that the genetic operators alter the best chromosome's gene so that whatever was good about them is destroyed.

To improve the properties of roulette wheel reproduction, an alternative called steady state reproduction is used in this paper. In this method,  $P_r$  percent of the best individuals will just be copied into the new generation, while the remained  $(100 - P_r)$  percent individuals will be treated by the GA operators. This method removes the drawbacks in roulette wheel, the best individuals will always reproduce because they are simply copied into the new generation and their genes are not changed by GA operators. The factor  $P_r$  is set between 5% - 10% to guarantee the number of remaining individuals large enough and to yield enough new individuals to compete in the new generation.

## 2.4. Crossover

Crossover is one of important operators in GA unlike the case in evolutionary programming (EP), in which crossover is not implemented and only mutation is

carried out.

There are different possible crossovers in GA literature, e.g. one point crossover and two point crossover. For one point crossover, one part of the parent chromosomes are exchanged at the randomly selected point, another part is kept the same as before. The simplicity of one point crossover makes it be widely used in GA operators, but its drawback is obvious, that is the possible searching space is limited because only one point is chosen.

In this paper, we use the two point crossover to overcome the shortage of the previous one. Fig. 4a and Fig. 4b illustrate the concept of this method.

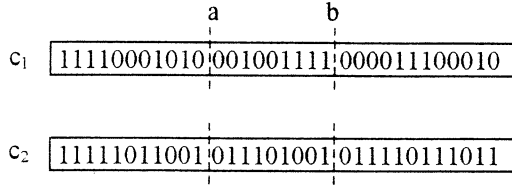


Fig.4a The parent chromosomes

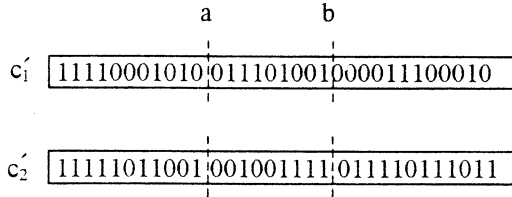


Fig. 4b The children chromosomes after crossover

In Fig. 4a,  $c_1$  and  $c_2$  are two parent chromosomes chosen to crossover. The bits of  $c_1$  and  $c_2$  between  $a$  and  $b$  will exchange each other, while the other bits in the parent chromosome keep unchanged. The children chromosomes after crossover are illustrated by Fig. 4b.

### 2.5. Mutation

Mutations are important for keeping a bit of wildness and random search flavor to optimization process. In this paper, we use the general random mutation method, that means each gene in a chromosome changes its value (called allele) from '0' to '1', or from '1' to '0' at the given probability  $P_m$ , where  $P_m$  is set to 0.0025 in the following simulation.

## 3. SIMULATION

### 3.1. Training and Testing Data

The temperature control system being approximated is multi-input one output nonlinear system illustrated by Fig.5a - Fig. 5f. Where,  $x_1, x_2, x_3, x_4, x_5$ , are inputs of neural networks,  $y$  is the output the same network. There are 666 sets of input and output data as the real inputs and output sets. Two parts of these sets are separated, one is called the training data, another is called testing data. The training data is used to train the neural network and the testing data is used calculate the fitness of the same neural network to sort the networks and find the best one in one generation. Every two real data in time sequence compose of the training sets, with

the remaining 333 real data compose the testing data. Each generated neural network by GA operator or randomly at start stage will be fully trained using the training data and then, its fitness will be calculated and to compare the best 10 existing networks. If its fitness is better (usually larger) than any of the best 10 networks, it will replace one of the older 10 best networks in the associated position. Otherwise, it will be discarded.

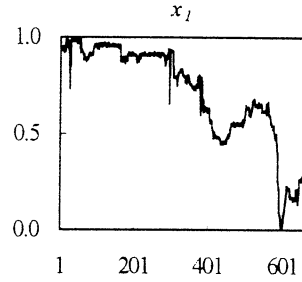


Fig.5a

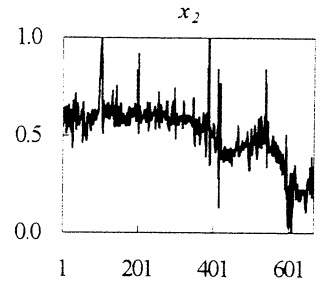


Fig.5b

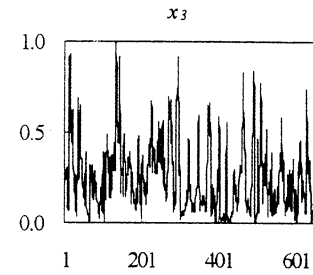


Fig.5c

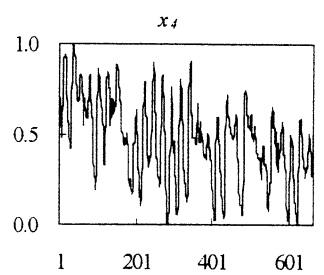


Fig.5d

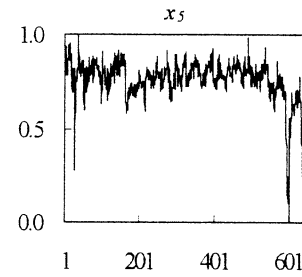


Fig.5e

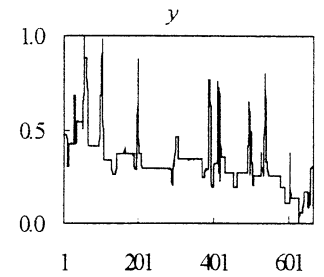


Fig.5f

To consider the influence from both the training and testing data, a combined fitness is employed in estimating the feature of evolved neural network, see Eq.2.

$$J_f = \frac{\mu_1 J_{tr} + \mu_2 J_{te}}{2} \quad (2)$$

where  $J_{tr}$ ,  $J_{te}$  is the fitness of neural network based on the training data and testing data, respectively.  $\mu_1$  and  $\mu_2$  are the influence coefficients for training and testing data, respectively. In this paper, the coefficients are selected as  $\mu_1 = 0.2$ ,  $\mu_2 = 0.8$ . The fitness based on the testing data has more influence in the final estimation of neural network.

### 3.2. Data Pre-Processing

Data pre-processing is important for the neural

training. There are numerous data pre-processing algorithms for different purposes, e.g. the data linearization which employs some transform to make linearization of the original non-linear data. In this paper, we employ the general dimensionless normalization technique to pre-process the real data and form the training and testing data. The data normalization is performed by Eq. 3.

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (3)$$

where  $x'$ ,  $x$ ,  $x_{\min}$  and  $x_{\max}$  is the variable after normalization, the original variable, the minimum of  $x$  and the maximum of  $x$ , respectively. After the data normalization, the value of training and testing data will be limited to the range of 0 - 1, see Fig. 5a - Fig. 5f.

### 3.3. Simulation Result

The simulation was carried out using the real data illustrated by Fig. 5a - Fig. 5f. as training and testing data sets. The system parameters are set as following.

Population size  $N_p = 30$ , generation number  $N_g = 100$ , the input factor  $\alpha = 0.05$ , the hidden neuron factor  $\beta = 0.2$ , reproduction factor  $P_r = 0.06$ , the mutation probability  $P_m = 0.002$ , using two point crossover method, the maximum number of hidden neurons  $N_{hmax} = 16$ , the maximum number of inputs  $N_{imax} = 5$ ,  $k_s = 700$ ,  $k_e = 100$  (the definition of  $k_s$  and  $k_e$  reference 2.2), the initial random weights are limited between  $\pm 0.3$ , the learning rate is 0.1, the momentum coefficient is 0.08. The final result are shown by Fig. 6 and Fig. 7.

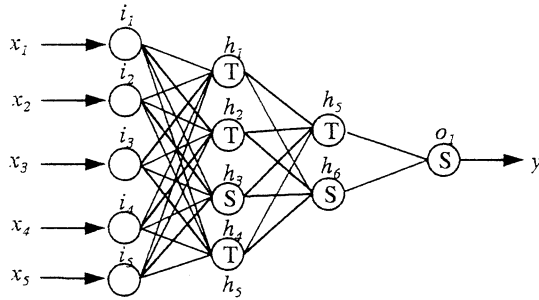


Fig. 6 The best network generated by GA

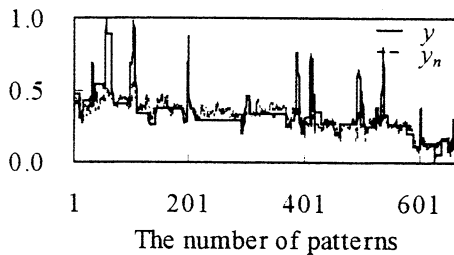


Fig. 7 Neural network output  $y_n$  comparison with the system real output  $y$ .

Fig. 6 is the best neural network evolved automatically by genetic algorithm with 5 inputs, 6 hidden neurons and one output.

Fig. 7 illustrates the neural network output  $y_n$  comparison with the system real output  $y$ . This network is evolved in 49 generation and is trained 8921 steps, with the training average absolute error  $\delta_{tr} = 0.0561$ , testing average absolute error  $\delta_{te} = 0.0581$ , the fitness of this network  $J_f = 9.19$ .

## 4. CONCLUSION

We make some attempts to automatically search for the optimal architecture of neural networks employing genetic algorithm. The simulation results show that the presented method is suitable in nonlinear function approximation application. According to the simulation result, a feedforward neural network with 6 hidden neurons may get the average absolute error about 0.05. In the near future, we will add other kinds of neural networks and training algorithms to expand the searching ability of this method. Also, we are considering to use this method to other applications.

## REFERENCES

- [1] Parisi R, Claudio EDD, Orlandi G and Rao BD (1996), A generalized learning paradigm exploring the structure of feedforward neural networks, IEEE Transactions on Neural Networks, 7(6):1450-1459.
- [2] Baldi PF and Hornik K (1995), Learning in linear neural networks: A survey, IEEE Transactions on Neural Networks, 6(4):837- 858.
- [3] Webb AR (1994), Functional approximation by feedforward networks: A least squares approach to generalization, IEEE Transactions on Neural Networks, 5(3):363- 371.
- [4] Sugisaka M and Liu ZJ (1998), The simulation of artificial neural network methods in cooler control using neurocomputer, Proceeding of the Third Workshop of Int. Institute for General Systems Studies, Qinhuangdao, China, July 26-28, 1998, pp.140-143.
- [5] Sugisaka M and Liu ZJ (1998), The application of neurocomputer for controlling the temperature of cooler of sinter plant, Proceeding of XIII Int. Conference on System Science, Wroclaw, Poland, Sept. 15-18, 1998, Vol. III, pp. 352-358.
- [6] Sugisaka M and Liu ZJ (1998), Artificial neural network and application in temperature control system, Proceeding of Korea Automatic Control Conference (KACC '98), Pusan, Korea, Oct. 15-17, 1998, pp. 260-264.
- [7] Yao X and Liu Y (1997), A new evolutionary system for evolving artificial neural networks, IEEE Transactions on Neural Networks, 8(3):694-713.
- [8] Castellano G, Fanelli AM and Pelillo M (1997), An iterative pruning algorithm for feed-forward neural networks, IEEE Transactions on Neural Networks, 8(3):519-531.

## Author Index

<b>A</b>		<b>D</b>	
Abe, K.	484, 488, 492, 496, 500, 737, 741	Dai, R.	I-27
Adachi, M.	82	Das, S.	150
Adachi, S.	415	<b>E</b>	
Aihara, K.	78, 82, 86, 90, 98	Eggenberger, P.	324
Aiyoshi, E.	130, 150	Endo, I.	34
Akama, S.	626	Endo, S.	391, 395, 407
Akiyama, N.	759	Eom, T.-D.	228, 763
Akizuki, K.	602	Etani, T.	16
An, S.-K.	179	<b>F</b>	
Ando, H.	682	Fei, Y.	I-27
Aoki, T.	291	Filliat, D.	745
Arai, F.	512	Fredslund, J.	I-9
Arai, K.	150	Fujii, N.	166
Aramaki, S.	266	Fujii, T.	34
Arendt, J.A.	I-9	Fujinaka, T.	146
Asai, T.	305, 309	Fujisawa, K.	291
Asama, H.	34, 50	Fujiwara, Y.	61, 65, 250,
Ascott, R.	12	Fukuda, T.	504, 512
Asharif, M.R.	403	Furuhashi, T.	508
<b>B</b>		<b>G</b>	
Baba, T.	28	Gao, D.-Z.	662
Bae, J.-I.	170	Garis, H.D.	606, 610
Baek, S.-M.	658	Gen, M.	448, 452, 456, 460
Belew, R.K.	OP-22	Gers, F.	606, 610
Bubnicki, Z.	OP-34	Goto, S.	630
<b>C</b>		Goto, Y.	200, 204, 208, 283
Casti, J.L.	OP-15	Guo, Q.	274
Chakarborty, I.	715	<b>H</b>	
Chang, M.	536	Hagan, S.	480, 588
Cho, D.-Y.	236	Hagiwara, M.	138
Cho, S.-B.	297	Han, M.-C.	175
Choi, J.	448	Han, S.-H.	170, 188
Choi, J.W.	188	Han, S.-J.	179
Chung, M.-J.	650	Hanazaki, I.	602
Colgate, S.	OP-21	Harada, H.	375

Hart, J.	I-1	Ishinishi, M.	524
Haruyama, M.	122	Islam, M.M.	305, 309
Hashem M.M.A.	516, 618	Isogai, M.	512
Hashimoto, H.	28, 40	Itiki, C.	313, 321
Hashimoto, S.	504	Itoh, K.	719
Hatanaka, T.	301	Ivan, L.	480
Hatano, I.	166	Iwakura, H.	544
Hattori, K.	362	Iwamoto, T.	58
Hayakawa, S.	291	Izumi, K.	516, 520, 618, 626
Hayashi, H.	90		
Hidai, K.	283		J
Hirafuji, M.	480, 588	James, D.J.G.	I-2
Hirasawa, K.	694, 699	Jiang, R.	I-31, 472
Hirashima, Y.	118	Jinnai, H.	568
Hirata, Y.	50	Jo, Y.-G.	220
Hirayama, H.	126, 733	Jun, H.-B.	232
Hirayama, T.	358	Jung, M.-J.	670
Hiroi, R.	560		
Hokari, S.	344		K
Hong, S.-G.	763	Kaetsu, H.	34, 50
Honma, N.	741	Kakazu, Y.	350
Hori, G.	86	Kalaba, R.	313, 321
Hori, K.	464	Kamano, T.	614
Horie, R.	130	Kambara, T.	362, 468, 644
Horio, Y.	90	Kang, H.	216, 220
Hough, M.	606, 610	Kashimori, Y.	362, 468, 644
Hu, D.	I-31, 472	Katai, O.	674
Hu, J.	694	Kataoka, Y.	614
Hu, Z.	46	Kato, Y.	636
Huang, J.	138	Kawabata, K.	50
Hwang, G.-H.	183	Kawaji, S.	46
		Kawakami, K.	154
	I	Kawata, S.	122
Ichikawa, A.	719	Keller, E.L.	150
Ichiki, T.	576	Kim, D.-Y.	650
Ida, K.	448, 456, 460	Kim, H.-S.	670
Ikeguchi, T.	94	Kim, J.-H.	670
Imai, M.	16	Kim, J.R.	460
Inaba, M.	737	Kim, M.-S.	763
Inabayashi, S.	636	Kim, S.	192
Inabayashi, S.	707	Kim, S.-I.	576, 584
Inoue, A.	118	Kimura, H.	OP-4
Ishida, Y.	69, 419	Kimura, K.	371
Ishiguro, A.	324	Kimura, T.	196
Ishii, H.	682	Kinjo, H.	431, 435

Kinoshita, M.	246
Kinouchi, Y.	707
Kita, H.	330, 334
Kitamura, S.	162, 287
Kitazawa, M.	371, 375
Kitazoe, T.	568, 576, 584
Kiyooka, T.	383
Kizu, S.	415
Kobayashi, I.	90
Kobayashi, S.	334
Kodama, S.	114
Kodjabachian, J.	745
Koga, H.	556
Kohashi, T.	340
Kohata, N.	28
Kojima, F.	504
Kojima, M.	751
Konar, A.	690, 715
Kondo, H.	711
Kondo, T.	324
Korkin, M.	606, 610
Kosaka, T.	142
Kosuge, K.	50
Kryssanov, V.V.	162
Kubo, M.	751
Kubota, N.	504
Kuc, T.-Y.	658
Kumagai, S.	106
Kunii, Y.	40
Kuribayashi, D.	34
Kuribayashi, K.	440, 444
Kurono, S.	266

## L

Lachner, K.	OP-21
Lee, C.-Y.	763
Lee, D.-W.	224, 232
Lee, D.Y.	188
Lee, I.-J.	658
Lee, J.-J.	I-16, 228, 279, 763
Lee, J.-M.	179, 188
Lee, M.C.	188
Lee, M.H.	170, 175, 179, 183, 188, 192
Lee, S.-Y.	699

Lehnert, B.	OP-21
Lei, X.-Y.	379
Li, D.	682
Li, F.	755
Li, S.-R.	755
Li, Y.-Z.	448, 456
Liang, X.	666
Liu, Z.-J.	767
Lu, B.	694
Lu, Q.	755
Luisi, L.	OP-21
Lund, H.H.	I-9
Luo, Y.	I-31, 472

## M

Maekawa, S.	61, 65
Maeshiro, T.	258
Manabe, K.	572, 580
Mandal, A.K.	690, 715
Mautner, C.	OP-22
Masuda, S.	118
Matrosov, V.M.	399
Matrossov, I.V.	411
Matsumoto, G.	OP-7
Mayer, B.	OP-21
Meyer, J.-A.	745
Mignonneau, L.	73
Mikami, S.	751
Min, S.-K.	216
Minamitani, H.	134
Mishima, T.	200, 204, 208, 212, 283
Miyagi, H.	395, 407
Miyagi, H.	439
Miyake, T.	379
Miyamoto, T.	106
Miyamoto, Y.	266
Miyano, T.	98
Mizoguchi, H.	200, 204, 208, 212, 283
Mizuhara, H.	363
Mizutani, M.	707
Mochizuki, M.	134
Mori, N.	330
Mun, K.-J.	183

Munekata, M.	358	Ohkura, K.	154, 536
Murakami, K.	444	Ohnishi, K.	344
Murakoshi, T.	602	Ohuchi, A.	54, 340, 354, 358
Murao, H.	287		759
Murase, K.	305, 309	Okamoto, T.	69
Murata, J.	694	Okazaki, K.	423
Murata, T.	354	Okita, Y.	126, 733
		Okuma, S.	291
	N	Omatsu, S.	142, 146
Nagata, F.	626	Ono, N.	540
Nakagawa, M.	552, 556, 560	Ono, T.	16
Nakagawa, T.	737	Oshima, T.	OP-5
Nakagawa, T.	682	Oura, K.	602
Nakajima, H.	106	Ozaki, T.	102
Nakajima, R.	196	Ozawa, K.	484
Nakamura, A.	275		
Nakamura, M.	630		P
Nakanishi, M.	427	Pagliarini, L.	I-9
Nakano, K.	723	Park, H.-K.	650
Nakasuka, S.	464	Park, J.-H.	183
Nakatsu, R.	5	Park, K.-B.	279
Namatame, A.	524, 528, 532	Patnaik, S.	690
Nanayakkara, D.T.	520		
Narita, Y.	362		Q
Natsuyama, H.	313, 321	Qu, C.	662
Nawa, N. E.	606, 610		
Nerome, M.	395		R
Nian, W.	423	Rahman, S.B.A	711
Nicoud, J. D.	I-41	Rasmussen, S.	OP-21
Niino, Y.	648	Ray, T.	I-1
Nishi, R.	439	Riquimaroux, H.	572, 580
Nishikawa, Y.	330, 363	Rui, F.	666
Nishimura, T.	488	Rumchev, V.	I-2
Nobuki, O.	387	Ryan, C.	480
Nordahl, M.G.	598		
Nordin, P.	598		S
Numata, M.	500	Sakai, Y.	371
		Sakamoto, T.	444
	O	Samadi, S.	544
Ochi, H.	435	Sanchez, E.	240
Ochi, Y.	644	Sanefuji, K.	622
Odagiri, R.	305, 309	Sano, A.	686
Ogasawara, T.	275	Sato, K.	626
Ohashi, S.	350	Satou, A.	707
Ohkawa, K.	24	Sawai, H.	415

	V	Yoshii, S.	350
Vadakkepat, P.	670	Yoshikawa, H.	682
	W	Yoshimi, T.	592
Wada, M.	246, 751	Yoshioka, M.	146
Wakamatsu, H.	703		Z
Wang, X.	654	Zhang, B.-T.	236
Wang, X.	729	Zhang, C.	662
Watanabe, K.	516, 520, 618, 622, 626	Zhang, X.	703
Watanabe, M.	78	Zhang, Y.-G.	I-22
Wei, Y.	305, 309	Zheng, M.-H.	468
Wells, W. R.	OP-28	Zhou, G.	452
Whitten, D.	OP-21		
Wu, J.-L.	363, 367, 371, 375, 379, 383, 387		
Wu, W.	682		
Wu, X.	I-22		
	X		
Xi, H.	472		
Xu, X.	666		
	Y		
Yamada, K.	391, 395, 407		
Yamada, K.	158		
Yamada, S.	484, 496, 500		
Yamaguchi, A.	751		
Yamaguchi, T.	28		
Yamamoto, H.	270		
Yamamoto, M.	354, 358, 759		
Yamamoto, T.	431, 435		
Yamashita, T.	54		
Yamauchi, S.	407		
Yanagawa, H.	344		
Yasuno, T.	614		
Yim, W.	OP-28		
Yoneyama, J.	719		
Yonezu, M.	427		
Yoshida, H.	367		
Yoshida, S.	540		
Yoshida, T.	427		
Yoshida, Y.	614		
Yoshihara, I.	484, 488, 492, 496, 500, 737		



Sawaragi, T. 674  
 Sayama, H. 254  
 Sekine, O. 196  
 Shi, Z. 102  
 Shiba, T. 358  
 Shibata, H. 544  
 Shibata, T. 20, 24  
 Shibuta, H. 98  
 Shigehara, T. 200, 204, 208, 212,  
 283  
 Shii, T. 568  
 Shim, H.-S. 670  
 Shimoda, H. 682  
 Shimohara, K. 464  
 Shinomiya, Y. 196  
 Shimoyama, Y. 532  
 Shin, W.-J. 699  
 Sim K.-B. 224, 232  
 Sohn, K.-O. 658  
 Sommerer, C. 73  
 Son, K. 188  
 Song, G.-B. 297  
 Solem, J. OP-21  
 Solomides, A. 5  
 Stilman, B. I-35  
 Suehiro, T. 275  
 Sugawara, K. 488, 492, 496, 500,  
 737  
 Sugi, T. 630  
 Sugisaka, M. I-22, 648, 654, 729,  
 767  
 Suzuki, H. 118  
 Suzuki, K. 54, 340, 354, 358  
 759  
 Suzuki, T. 614  
 Suzuki, T. 291  
 Suzuki, Y. 262  
 Svinin, M. M. 158  
 Shouji, F. 707

# T

Tabuchi, F. 431  
 Tabuse, M. 548  
 Takadama, K. 464  
 Takahashi, O. 334

Takahide, M. 28  
 Takai, S. 110  
 Takayanagi, T. 340  
 Takeda, A. 496  
 Takeda, H. 741  
 Takeo, K. 50  
 Takeuchi, I. 508  
 Takiya, S. 358  
 Takuno, M. 301  
 Tamaki, H. 162  
 Tamura, S. 423, 636  
 Tamura, S. 301  
 Tamura, Y. 102  
 Tanaka, A. 114  
 Tanaka, H. 244, 262  
 Tanaka-Yamawaki, M. 548  
 Tang, J. 440, 622  
 Tanie, K. 20, 24  
 Taura, T. 592  
 Taylor, C. OP-1  
 Terano, T. 464  
 Teshiba, M.W. 204, 212  
 Tilden, M. OP-2  
 Toma, N. 391  
 Toma, T. 403  
 Tomizuka, M. 723  
 Tosa, N. 1, 5  
 Toyoda, S. 640  
 Tsuchiya, K. 320  
 Tsuji, T. 279  
 Tsujita, K. 320  
 Tsukune, H. 275  
 Tsutsumi, K. 358

# U

Uchikawa, Y. 324  
 Uchimura, K. 46  
 Uchiyama, H. 564  
 Ueda, K. 154, 158, 166, 536  
 Uezato, E. 431  
 Uno, K. 528  
 Uosaki, K. 301  
 Uribe, A.P. 240  
 Ushio, T. 114