

PROCEEDINGS OF THE FIFTH INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS

(AROB 5th '00)

Vol.2

Jan. 26-Jan. 28, 2000
Compal Hall, Oita, JAPAN

Editors : Masanori Sugisaka and Hiroshi Tanaka
ISBN4-9900462-0-X

Proceedings of The Fifth International Symposium on

ARTIFICIAL LIFE AND ROBOTICS

(AROB 5th '00)

for Human Welfare and Artificial Life Robotics

January 26-28, 2000
Compal Hall, Oita, JAPAN

Editors: Masanori Sugisaka and Hiroshi Tanaka

ISBN4-9900462-0-X, 2000

THE THIRD INTERNATIONAL SYMPOSIUM ON ARTIFICIAL LIFE AND ROBOTICS (AROB 5th '00)

ORGANIZED BY

Oita University
Under The Sponsorship of Ministry of Education,
Science, Sports, and Culture, Japanese Government

CO-SPONSORED BY

Santa Fe Institute(SFI, USA)
The Institute of Control, Automation, and Systems Engineers(ICASE, Korea)
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
Beppu Municipal Government
Kyushu Bureau of International Trade and Industry, MITI
NHK Oita Station
Nihonkeizai Shimbun
Nikkan Kogyo Shimbun
Oita Asahi Broadcasting
OBS Broadcast Company
Oita Chamber of Commerce and Industry
Oitagodo Shinbunsya
Oita Industrial Group Society
Oita Municipal Government
Oita Prefectural Government
Oita System Control Society
Science and Technology Agency
TOS Broadcast Company

HONORARY PRESIDENT

M.Hiramatsu (Governor, Oita Prefecture)

ADVISORY COMMITTEE CHAIRMAN

M.Ito (Director, RIKEN, Japan)

GENERAL CHAIRMAN

M.Sugisaka (Oita Unviersity, Japan)

VICE CHAIRMAN

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

PROGRAM CHAIRMAN

H.Tanaka (Tokyo Medical & Dental University, Japan)

ADVISORY COMMITTEE

S.Fujimura (The University of Tokyo, Japan)
M.Ito (Director, RIKEN, Japan) (Chairman)
H.Kimura (The University of Tokyo, Japan)
S.Ueno (Kyoto University, Japan)

ORGANIZING COMMITTEE

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

W.R.Wells (University of Nevada-Las Vegas, USA)
Y.G.Zhang (Academia Sinica, China)

STEERING COMMITTEE

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

PROGRAM COMMITTEE

K.Abe (Tohoku University, Japan)
K.Aihara (The University of Tokyo, Japan) (Co-chairman)
M.Bedau (Reed College, USA)
R.Belew (University of California-San Diego, USA)
Z.Bubunicki (Wroclaw University of Technology, Poland)
T.Christaller (CMD-German National Research Center for Information Technology, Germany)
T.Fujii (RIKEN, Japan)
M.Gen (Ashikaga Institute of Technology, Japan)
T.Gomi (AAI, Canada)
I.Harvey (University of Sussex, UK)
H.Hashimoto (The University of Tokyo, Japan) (Co-chairman)
H.Hirayama (Asahikawa Medical College, Japan)
P.Husbands (University of Sussex, UK)
K.Ito (Tokyo Institute of Technology, Japan)
J.Johnson (The Open University, UK)
Y.Kakazu (Hokkaido University, Japan)
R.E.Kalaba (University of Southern California, USA)
H.Kashiwagi (Kumamoto University, Japan)
O.Katai (Kyoto University, Japan)
S.Kawata (Tokyo Metropolitan University, Japan)
J.H.Kim (KAIST, Korea)
S.Kitamura (Kobe University, Japan)
H.Kitano (Sony Computer Science Laboratory Inc., Japan)

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

LOCAL ARRANGEMENT COMMITTEE

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

TOPICS

Hardware Oriented Topics are welcome
in the fields given by

Artificial Brain Research
Artificial Intelligence
Artificial Life
Artificial Liferobotics
Artificial Living
Artificial Mind Research
Brain Science
Chaos
Cognitive Science
Complexity
Computer Graphics
Evolutionary Computations
Fuzzy Control
Genetic Algorithms
Innovative Computations
Intelligent Control and Modeling
Micromachines
Micro-Robot World Cup Soccer Tournament
Mobile Vehicles
Molecular Biology
Neural Networks
Neurocomputers
Neuro Computing Technologies
and its Application for Hardware
Robotics
Robust Virtual Engineering
Virtual Reality
Related Fields

COPYRIGHT

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

All correspondence relating to the symposium should be addressed to:

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

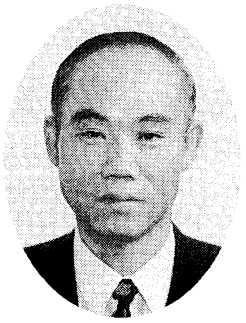
WWW Home Page http://AROB.cc.oita-u.ac.jp/

AROB 5th '00 is supported financially by the following companies

Oita Gas Co., Ltd.
Sanwa Shurui Co., Ltd.
Shintsurukai Kosan Co., Ltd.
Yatsushika Sake-Brewing Co., Ltd.

MESSAGE

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



It is my great honor to invite you all to The Fifth International Symposium on Artificial Life and Robotics(AROB 5th '00), 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, ICASE, SICE, RSJ, and IEEJ, Japan. This symposium invites you all to discuss development of new technologies concerning Artificial Life and Robotics based on simulation and hardware in twenty first century and in the new millennium. 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 and robotics has been expected in industries, business, etc. to contribute for human society. The special topic in AROB 5th '00 is "for Human Welfare and Artificial Life Robotics".

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 5th '00 will become a celebration to the establishment of our international joint research institute on artificial life and robotics for the new millennium by the support of Monbusho's program of center of excellence and etc. I hope that you will obtain fruitful results by exchanging ideas through discussions during the symposium and also will enjoy your stay in Oita.

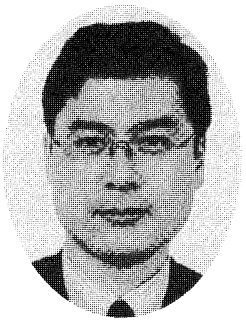
I am looking forward to meeting you in Oita.

Masanori Sugisaka

M. Sugisaka

January 12, 2000

MESSAGE



Keisuke Yoshio
Ministry of Education, Science, Sports and Culture (Monbusho)
(Director, International Scientific Affairs Division,
Science and International Affairs Bureau)

Ladies and Gentlemen:

It is my great pleasure to send my message to the Symposium participants, on behalf of the Ministry of Education, Science, Sports and Culture (Monbusho), at the opening ceremony of the Fifth (2000) Anniversary of International Symposium on Artificial Life and Robotics (AROB) in a new millennium.

First of all, I would like to express my hearty welcome to all participants here today, and particularly to those scientists who have come a long way to Japan. The purpose of this symposium is to provide the bases to the collaborative work on the theory of artificial life and its applied research on robotics through an exchange of views and information among distinguished scientists both from abroad and in Japan.

Nowadays research on artificial life is in the embryonic stage in our country, but its application in various fields, in particular in industrial society is gradually realized. I am certain that this symposium will be very productive and will fully facilitate such realizations through energetic exchanges of ideas about the new trends in computer science and robotics.

I also consider that this kind of symposium hold here in Japan will positively contribute to sending out our nation's excellent academic achievements and information worldwide and to enhancing academic cooperation in this field.

Monbusho is supporting such symposiums organized by Japanese university not only to stimulate the internationalization of universities and promote international exchanges of knowledge and collaborative research between researchers but also to assist in forming the centers of excellence through international contacts. We are honored to say that this is one such symposium.

Also, I would like to point out that Prime Minister, Mr.Obuchi's Cabinet has decided recently to develop research on artificial life and robotics as one of our national projects for a new millennium. Monbusho considers that this symposium is in line with the Cabinet decision.

In closing, I sincerely hope that this symposium will be a great success through your active discussions, and I would like to express my sincere gratitude to Professor Sugisaka and the staff members concerned who have made every effort to hold this symposium.

Finally, I hope that the participants from abroad will have many opportunities to learn about and enjoy Japanese culture and society, and will come to understand more about Japan.

Thank you very much.

K. Yoshio

January 12, 2000

MESSAGE

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



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

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

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

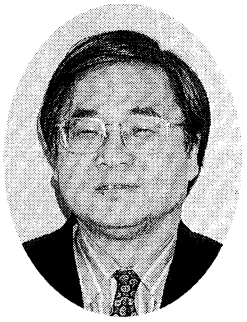
Since the very first Artificial Life meeting in 1987 in Los Alamos, New Mexico, the Santa Fe Institute (SFI) has been at the forefront of this shift in emphasis from matter to information. By the same token, SFI has actively supported such research initiatives in every corner of the world. This support has extended to the Artificial Life and Robotics meetings here in Japan, since the time of the very first meeting in 1996. Each year, researchers from the SFI faculty have come to Japan to meet with others at these AROB meetings, in order to present edge-of-the-frontier ideas and to exchange views on how the fields of ALife and robotics are progressing. So it is a great pleasure for me to again represent SFI on the Organizing Committee of AROB5, and to welcome everyone to this event.

John CASTI

J.L. Casti

January 12, 2000

MESSAGE



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

On behalf of the program committee, it is truly my great honor to invite you all to The Fifth International Symposium on Artificial Life and Robotics (AROB 5th '00). 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 Oita.

Hiroshi Tanaka

H. Tanaka

January 12, 2000

TECHNICAL PAPER INDEX

Plenary Lectures (Plenary Talks)

P1 *Intelligent systems -a new horizon of science and technology-* ……*P-1*
F. Harashima (Tokyo Metropolitan Institute of Technology, Japan)

P2 *Brain-style information systems ---mathematical foundations* ……*P-2*
S. Amari (RIKEN, Japan)

P3 *Toward universal laws of evolving systems* ……*P-3*
M. A. Bedau (Reed College, USA)

Banquet Talk

B1 *Simulation in business and geopolitics* ……*B-1*
J. L. Casti (Santa Fe Institute, USA)

Invited Lectures (Invited Talks)

IT1-1 *The application of learning algorithms and uncertain variables in the knowledge-based pattern recognition* ……*I-1*
Z. Bubnicki (Wroclaw University of Technology, Poland)

IT1-2 *Compression and adaptation* ……*I-6*
C. E. Taylor, T. Teal (University of California-Los Angeles, USA)

IT1-3 *A review to the international symposium on artificial life and robotics* ……*I-7*
M. Sugisaka (Oita University, Japan)
Y. G. Zhang (ATR, Japan)

IT1-4 *Measures of evolvability in tierra* ……*I-12*
T. S. Ray (Santa Fe Institute, USA)
C. Xu (ATR, Japan)

IT2-1 *The control structure of artificial creatures* ……I-16
D. Zhou, R. Dai (Chinese Academy of Science, China)

IT2-2 *Visual communication in swarms of intelligent robot agents* ……I-20
J. Johnson (The Open University, UK)

IT2-3 *Evolutionary on-line self-organization of autonomous robots* ……I-30
J. D. Floreano, J. Urzelai (Swiss Federal Institute of Technology, Switzerland)

WA1: Nonlinear Complex Dynamics on Medical Science (General Session) (Room 304)

WA1-1 *Prediction of significant oculocardiac reflex by nonlinear complex cardiac dynamics* …… 1
M. K. Yum (Hanyang University Kuri Hospital, Korea)
H. S. Kim (SNUH, Korea)

WA1-2 *Nonlinear analysis of EEG in patient with schizophrenia* …… 5
J. H. Chae, D. J. Kim, K. S. Kim, T. Y. Jun, W. M. Bahk
(The Catholic University of Korea, Korea)
J. Jeong (Yale University, U.S.A)
S. Y. Kim (Korea Advanced Institute of Science and Technology, Korea)

WA2: Healing Technology and Artificial Life (Organized Session) (Room 304)

WA2-1 *Healing technology with artificial life -unconscious flow-* …… 9
N. Tosa (ATR, Japan)

WA2-2 *Future communications based on KANSEI technologies* …… 13
R. Nakatsu (ATR, Japan)

WA2-3 *The artist and the automaton: evolving musics, a second nature, artists gently break the news...* …… 19
R. Berry (ATR, Japan)

WA2-4 *Integrating real and virtual worlds in shared space* ····· 22
I. Poupyrev (ATR, Japan)
M. Billinghurst, R. May (University of Washington, USA)
H. Kato (Hiroshima City University, Japan)

WA2-5 *Sensing, communication and intentionality in artificial life.* ····· 26
S. Jones (ATR, Japan)

WB1: Self-Reproduction and Self-Decomposition (Organized Session) (Room 305)

WB1-1 *Minimum density of functional proteins to make a system evolvable* ····· 30
H. Suzuki (ATR, Japan)

WB1-2 *Diffusible immortal ALife rarely exterminate diffusible mortal ALife in one finite, heterogeneous ecosystem.* ····· 34
T. Maekawa, K. Shimohara, E. Nishina, N. Kawai, T. Oohashi
(ATR, Japan)
O. Ueno (Gifu University School of Medicine, Japan)

WB1-3 *Evolutionary advantage of self-decomposition mechanism* ····· 40
T. Maekawa, K. Shimohara, E. Nishina, N. Kawai, T. Oohashi
(ATR, Japan)
O. Ueno (Gifu University School of Medicine, Japan)

WB1-4 *Proposal of a general simulator "SIVA" for heterogeneous environment and self-decomposable ALife.* ····· 45
T. Oohashi, T. Maekawa, E. Nihsina, N. Kawai, K. Shimohara
(ATR, Japan)
O. Ueno (Gifu University School of Medicine, Japan)

WB2: Smart Micro Mechatronics (Organized Session) (Room 305)

WB2-1 *Hybrid force and position control of miniature robotic fingers driven by piezoelectric actuators* 50
Z. Jiang (Yamaguchi University, Japan)
S. Chonan, Y. Namiki (Tohoku University, Japan)

WB2-2 *Development of a miniature spray robot using conduit guided wire for a Die casting process* 54
M. Hisano, Y. Takeuchi, K. Kuribayashi, T. Sakamoto, K. Murakami (Yamaguchi University, Japan)

WB2-3 *A navigation system for bending profile of Endoscope using strain gauges* 58
T. Tarumoto, K. Kuribayashi (Yamaguchi University, Japan)

WB2-4 *Development & estimation of dynamic characteristic model of sensor for sensing stiffness* 62
T. J. Chung (Kunsan National University, Korea)
D. P. Hong, H. Y. Kang, S. Y. Hwang, Y. S. Ku
(Chonbuk National University, Korea)

WB2-5 *Characteristics of a planar type micro electro magnetic sensor for inspection of metal cracks* 66
X. Ding, T. Hashida, K. Kuribayashi (Yamaguchi University, Japan)

WB3: Human Centered Systems (Organized Session) (Room 305)

WB3-1 *Human motion recognition in intelligent space* 70
T. Yamaguchi, J. H. Lee, H. Hashimoto (The University of Tokyo, Japan)

WB3-2 *Realization of a mood congruency effects based on a model of knowledge, emotion and intention* 76
T. Hashimoto (Nasu University, Japan)

WB3-3 *Human centered ITS* ····· 80
T. Yamaguchi, H. Nitta, K. Hirayama (Utsunomiya University, Japan)
H. Hashimoto (The University of Tokyo, Japan)
T. Takagi (Meiji University, Japan)

WB3-4 *Diversity oriented evolutionary parallel computation based on cultural background* ····· 84
N. Kohata, T. Yamaguchi, T. Baba (Utsunomiya University, Japan)
H. Hashimoto (The University of Tokyo, Japan)

WB3-5 *Visualization of patterns and construction of fuzzy inference system for integration of patterns and symbols* ····· 88
Y. Hattori, T. Furuhashi (Nagoya University, Japan)

WB3-6 *Performance of information recommendation agent based on reinforcement learning in society* ····· 92
N. Ishitani, T. Kinone, S. Tano
(University of Electro-Communications, Japan)

WB3-7 *Agent systems driven by intentional reasoning* ····· 96
T. Takagi, M. Mukaidono (Meiji University, Japan)
T. Yamaguchi (Utsunomiya University, Japan)

WB4: Genetic Algorithms I (General Session) (Room 305)

WB4-1 *Generalization ability of minimum description length for multiple-organisms learning by genetic programming* ·····100
A. Yoshida (Nara Institute of Science and Technology, Japan)

WB4-2 *Emergent design of optimal postures for mobile manipulators by using a Java interactive GA viewer* ·····104
K. Tagawa, H. Haneda (Kobe University, Japan)
M. H. Lee, J. M. Lee (Pusan National University, Korea)

WB4-3 *Effects of population topology in distributed parallel genetic algorithms* ·····108
D. Hu, R. Jiang, Y. Luo (Tsinghua University, China)

WB4-4 *EcoGA: maintaining diversity through adaption in genetic algorithms* ……112
L. Sheehan, J. J. Collins, M. Eaton (University of Limerick, Ireland)

WC1: Artificial Life I (General Session) (Room 309)

WC1-1 *Design of autonomous mobile robot action selector based on a learning artificial immune network structure* ……116
D. J. Lee, M. J. Lee, Y. K. Choi, S. Kim (Pusan National University, Korea)

WC1-2 *The clear explanation of different adaptive behaviors between Darwinian population and Lamarckian population in the changing environment* ……120
K. Yamasaki, M. Sekiguchi
(Tokyo University of Information Sciences, Japan)

WC1-3 *PICO_SCAN -using body data to create artificial life forms* ……124
L. Mignonneau, C. Sommerer (ATR, Japan)

WC1-4 *Proposal of a function discovery system using artificial life model and its applications* ……128
S. Serikawa, T. Shimomura (Kyushu Institute of Technology, Japan)

WC1-5 *Prediction of moving object position in visual perception system of artificial life* ……132
W. Nian, K. Okazaki (Fukui University, Japan)
S. Tamura (Osaka University, Japan)

WC2: Neural Network, Recognition and Behavior Control (Organized Session) (Room 309)

WC2-1 *Application of stereovision neural network to continuous speech recognition* ……137
T. Ichiki, T. Kitazoe (Miyazaki University, Japan)

WC2-2 *Verbal transformation effect by two and three layered neural network* ……141
M. Funamori, T. Kitazoe (Miyazaki University, Japan)

WC2-3 *Control system for Khepera robot by neural network with competition and cooperation* ……145
K. Sugihara, M. Tabuse (Miyazaki University, Japan)

WC2-4 *Pattern recognition of emotional states using voice, face image and thermal image of face* ……149
S. I. Kim, Y. Yoshitomi, T. Kitazoe (Miyazaki University, Japan)

WC2-5 *Quantitative analyses and fractal structures of fish school movements* ……153
T. Shinchi, T. Kitazoe, M. Tabuse (Miyazaki University, Japan)
H. Nishimura (Hyogo University, Japan)

WC3: Chaos Engineering (Organized Session) (Room 309)

WC3-1 *Contextual modulation with reinforcement learning in a temporal coding neural network* ……157
M. Watanabe, K. Aihara (The University of Tokyo, Japan)

WC3-2 *A simplified single neuron model as a rapid depolarization detector based on the H-H type neuron* ……161
Y. Sakumura, K. Aihara (The University of Tokyo, Japan)

WC3-3 *Effect of small random noise on the chaotic behavior of the Hodgikin-Huxley equations* ……165
H. Tanaka (CREST, Japan)
K. Aihara (The University of Tokyo, Japan)

WC3-4 *A mixed analog / digital circuit for quadratic assignment problems* ……169
H. Hayashi, Y. Horio (Tokyo Denki University, Japan)
K. Aihara (The University of Tokyo, Japan)

WC3-5 *A switched-capacitor chaotic neuro-computer: medium-scale implementation* ……173
M. Yoneda, I. Kobayashi (Tokyo Denki University, Japan)
K. Aihara (The University of Tokyo, Japan)

WC3-6 *Inter-event interval reconstruction of chaotic dynamics* ……177
I. Tokuda (Muroran Institute of Technology, Japan)
K. Aihara (The University of Tokyo, Japan)

WC3-7 *On measuring distances between strange attractors* ……181
T. Ikeguchi (Science University of Tokyo, Japan)

WC3-8 *Chaotic EEG analysis and judgment of brain death* ……185
G. Hori (RIKEN, Japan)
K. Aihara (The University of Tokyo, Japan)
Y. Mizuno, Y. Okuma (Juntendo University, Japan)

WC3-9 *Exploring dynamics of interest rate by nonlinear time series approaches* ……189
Z. Shi, Y. Tamura, T. Ozaki
(The Institute of Statistical Mathematics, Japan)

WC4: Artificial Brain (Organized Session) (Room 309)

WC4-1 *CoDi technique: cellular automata as neural networks* ……192
A. Buller, K. Shimohara (ATR, Japan)
T. Chodakowski (Gdansk University, Poland)
H. Hemmi (NTT, Japan)

WC4-2 *An interactive mechanism for CAM-brain* ……196
H. Hemmi, T. Hikage, K. Shimohara (NTT, Japan)

WC4-3 *Multi-agent based neural network as a dynamical brain model* ……200
S. Ohtani, K. Shimohara (ATR, Japan)
M. Ishido (Kobe University, Japan)

WC4-4 *Does the butterfly effect take place in human working memory?* ……204
A. Buller, K. Shimohara (ATR, Japan)

WC4-5 *Axonal growth in evolutionary neurogenesis* ……208
P. Eggenberger (ATR, Japan)

WD1: Genetic Algorithms II (General Session) (Room 302)

WD1-1 *Optimization of neural network structure based on a hybrid method* ……212
M. Sugisaka, Z. Liu (Oita University, Japan)

WD1-2 *The optimization of a neural network by genetic algorithm* ……216
M. Sugisaka, M. Itou (Oita University, Japan)

WD1-3 *Search optimum in multi-peak goal function space by genetic algorithm* ……220
M. Sugisaka, M. Shono (Oita University, Japan)

WD1-4 *The optimal design of the fuzzy membership function which used genetic algorithm* ……224
M. Sugisaka, T. Kurimoto (Oita University, Japan)

WD2: Complexity I (General Session) (Room 309)

WD2-1 *A new approach of self-organization -from the DNA point of view* ……228
Y. Zhang, G. Cao, X. Zhu (Shanghai Jiaotong University, China)

WD2-2 *Scale dependence of fractal dimensions in the human retinal blood vessel* ……232
M. Monma, Y. Yonezawa (Ibaraki University, Japan)
T. Igarashi (Ibaraki Prefectural University of Health Sciences, Japan)

WD2-3 *Mathematical analysis on an immune system* ……238
M. Kawakita, S. Shin (The University of Tokyo, Japan)

WD2-4 *Adaptive iteration learning control of upper limb motion with functional neuromuscular stimulation (FNS)* ……242
H. Wu, Z. Zhou, S. Xiong, G. Zhang (Tsinghua University, China)

WD3: Evolutionary Computations I (General Session) (Room 302)

WD3-1 *Visualisation of evolutionary algorithms using principal components analysis* ……247
J. J. Collins, M. O'Neill (University of Limerick, Ireland)

WD3-2 *Comparison between synchronous and asynchronous implementation of parallel genetic programming* ……251
S. Tongchim, P. Chongstitvatana (Chulalongkorn University, Thailand)

WD3-3 *A multi-agent reinforcement learning method with the estimation of the other agent's actions* ……255
Y. Nagayuki, S. Ishii, M. Ito
(Nara Institute of Science and Technology, Japan)
K. Shimohara, K. Doya (ATR, Japan)

WD3-4 *How learning can affect the course of evolution in dynamic environments* ……260
R. Suzuki, T. Arita (Nagoya University, Japan)

WD3-5 *Evolving a classifier system reward policy* ……264
T. Sepulveda, M. R. Gomes (INESC, Portugal)

WD3-6 *Extending prediction term of GP-based time series model* ……268
I. Yoshihara, T. Aoyama (Miyazaki University, Japan)
M. Numata, K. Abe (Tohoku University, Japan)
M. Yasunaga (University of Tsukuba, Japan)

WD4: Intelligent Control and Modeling I (General Session)(Room 302)

WD4-1 *Implementation of real-time control of assembly line on the back cover of a camera* ……272
M. H. Lee, K. Son, M.C. Lee, J. M. Lee, J. W. Choi, Y. H. Chang
(Pusan National University, Korea)
S. H. Han (Kyungnam University, Korea)

WD4-2 *Controlling conditional information by α-information* ……277
R. Kamimura (Tokai University, Japan)

WD4-3 *Complexity control method by stochastic analysis for recurrent neural networks* ……281
M. Sakai, N. Honma, K. Abe (Tohoku University, Japan)

WD4-4 *Control structure of stabilizing controllers for linear minimum phase single-input and single-output systems* ……285
K. Yamada (Yamagata University, Japan)

WD4-5 *Relation between saturation and settling time of antiwindup control u* ……289
left coprime factorization method
K. Yamada (Yamagata University, Japan)
Y. Funami (Shonai College of Industry and Technology, Jap

WD4-6 *Modeling of human motor control on manual tracking to moving visu* ……293
target based on neurophysiological background
T. Sugi, M. Nakamura (Saga University, Japan)
J. Ide (Seinan Gakuin University, Japan)
H. Shibasaki (Kyoto University, Japan)

TA1: Mobile Vehicles I (General Session) (Room 304)

TA1-1 *Path planning and navigation of a mobile robot as discrete optimizat oblems* ……297
H. Igarashi, K. Ioi (Kinki University, Japan)

TA1-2 *High-level spatial scenarios in WAVE* ……301
P. S. Sapaty (National Academy of Sciences of Ukraine, Ukraine)

TA1-3 *Towards a common set of experimental frameworks for the evaluation and* ……305
benchmarking of mobile robot control architectures
M. Eaton, J. J. Collins, L. Sheehan (University of Limerick, Ireland)

TA1-4 *An autonomous restoration of efficiency for AGV transportation system* ……309
T. Higashi (Murata Machinery, LTD., Japan)
K. Sekiyama (The Science University of Tokyo, Japan)
T. Fukuda (Nagoya University, Japan)

TA1-5 *Self-localization of a mobile robot by recognizing color objects in an omnidirectional* ……313
image
T. K. Kim, Y. J. Lee, M. J. Chung (KAIST, Korea)

TA1-6 *On-line scheduling of multiple mobile robot system using coordination protocol* ……317
D. H. Yoo, G. W. Chu, M. J. Chung (KAIST, Korea)

TA2: The State-of-The-Art of A-life Research in Korea (Organized Session) (Room 304)

TA2-1 *An on-line learning method for object-locating robots using genetic programming on evolvable hardware* ……321
H. S. Seok, K. J. Lee, J. G. Joung, B. T. Zhang
(Seoul National University, Korea)

TA2-2 *Generation of rules for swarm intelligence of autonomous mobile robots* ……325
S. K. Kim, S. G. Kong (Soongsil University, Korea)

TA2-3 *Evolvable cellular neural networks: a new paradigm for classifiers* ……329
H. Kang (Chung-Ang University, Korea)

TA2-4 *Applications of artificial life to developing robot and softbot* ……334
S. B. Cho (Yonsei University, Korea)

TA2-5 *Cooperative behavior of collective autonomous mobile robots based on online learning and evolution* ……338
D. W. Lee, H. B. Chun, K. B. Sim (Chung-Ang University, Korea)

TA3: Robotics I (Room 304) (General Session)

TA3-1 *Daydreaming robots* ……342
S. I. Ahson (King Saud University, Saudi Arabia)

TA3-2 *Four layered robot control architecture* ……347
J. J. Collins, L. Sheehan, M. Eaton, M. O'Neill
(University of Limerick, Ireland)

TA3-3 *Automatic generation of robot behaviours using grammatical evolution* ……351
M. O'Neill, J. J. Collins, C. Ryan (University of Limerick, Ireland)

TA3-4 *Behavior control of autonomous mobile robots using the block-based neural network* ……355
S. W. Moon, S. G. Kong (Soongsil University, Korea)

TA3-5 *From life to robotics: social robots* ……359
S. Reimann (GMD, Germany)
A. Mansour (RIKEN, Japan)

TA3-6 *Viewing autonomous robot design as self-organizing complex dynamical systems* ……363
A. D'Angelo (Udine University, Italy)
F. Montesello, E. Pagello (Padua University, Italy)

TA3-7 *Experiments of a floating underwater robot with 2 link manipulator* ……367
S. Sagara, T. Tanikawa, M. Tamura, R. Katoh
(Kyusyu Institute of Technology, Japan)

TA4: Robotics II & Fuzzy Control (General Session) (Room 304)

TA4-1 *A target following robot system built on an autonomous mobile platform* ……371
K. Ishikawa, A. Sakurai
(Japan Advanced Institute of Science and Technology, Japan)

TA4-2 *Eye-gaze tracking from sequential images* ……375
D. H. Kim, J. H. Kim, M. J. Chung (KAIST, Korea)

TA4-3 *Fuzzy system modeling based on subtractive clustering and generalized radial basis function network* ……379
I. H. Jo, K. B. Sim, H. Kang (Chung-Ang University, Korea)

TB1: Intelligent Systems (Organized Session) (Room 305)

TB1-1 *Human direct teaching of industrial articulated robot arms based on forceless control* ……383
D. Kushida, M. Nakamura, S. Goto (Saga University, Japan)
N. Kyura (Kinki University in Kyushu, Japan)

TB1-2 *Stable nonlinear controller design for Takagi-Sugeno fuzzy model* ……387
C. Y. Lee, T. D. Eom, J. J. Lee (KAIST, Korea)

TB1-3 *Generalized asymmetrical bidirectional associative memory for human skill transfer* ……391
T. D. Eom, J. J. Lee (KAIST, Korea)

TB1-4 *A new PID regulation controller with Lyapunov stability* ……395
J. H. Lee (Gyeongsang National University, Korea)

TB1-5 *Primary control strategy based on the idea of Abrain* ……399
M. Sugisaka (Oita University, Japan)
X. Wang (Oita Institute of Technology, Japan)

TB1-6 *Task priority control of redundant manipulators* ……402
Y. Choi, W. K. Chung (POSTECH, Korea)
S. R. Oh (KIST, Korea)

TB2: Toward Intelligence on Chip (Organized Session) (Room 305)

TB2-1 *Mobile robot control by sophisticated evolution of multiple CAM-Brains* ……406
G. B. Song, S. B. Cho (Yonsei University, Korea)

TB2-2 *Progressive evolution with subgoals that are common properties in elite population* ……410
M. Matsuzaki, T. Kawai, H. Ando, T. Shimada
(Nagoya University, Japan)

TB2-3 *A proposal of hardware GA engine compiler using extended C language and field programmable gate array* ……416
M. Fujii, T. Furuhashi (Nagoya University, Japan)

TB2-4 *A new genetic encoding for evolvable hardware - on the dilemma between unconstrained approach and effective search -* ……420
A. Nakata, M. Mizumoto (Osaka Electro-Communication University, Japan)

TB2-5 *Obstacle avoidance for autonomous mobile robot using anytime sensing* ……424
K. Fujisawa, T. Suzuki, S. Okuma (Nagoya University, Japan)
S. Hayakawa (Toyota Technological Institute, Japan)
T. Aoki (Nagoya Municipal Industrial Research Institute, Japan)

TB3: Interactive Self -Reflection (Organized Session) (Room 305)

TB3-1 *Interactive self-reflection model: aim and overview* ……430
K. Takadama, K. Shimohara (ATR, Japan)

TB3-2 *Learning is equal to participation: interactive acquisition of identity in a group* ……434
T. Shiose, T. Sawaragi, O. Katai (Kyoto University, Japan),
M. Okada (ATR, Japan)

TB3-3 *Interactive self-reflection model for communications* ……438
N. Okada, K. Shimohara, K. Takadama, M. Okada (ATR, Japan)
O. Katai (Kyoto University, Japan)

TB3-4 *Interactive self-reflection based reinforcement learning for multiagent coordination* ……442
T. Yamaguchi (Nara National College of Technology, Japan)

TB3-5 *On the efficiency of a decentralized continuous double auction market of locally aware agents -bridging the gap between economics and interactive self-reflection* ……446
N. E. Nawa, K. Takadama, K. Shimohara (ATR, Japan)
O. Katai (Kyoto University, Japan)

TB3-6 *Interactive self-reflection mechanism based on homogenized bipartite model* ……450
T. Maeshiro (ATR, Japan)
T. Morimoto, Y. Fujiwara (Kanagawa University, Japan)

TB3-7 *Agent architecture based on self-reflection learning classifier system* ……454
H. Inoue, K. Takadama, K. Shimohara, M. Okada (ATR, Japan)
O. Katai (Kyoto University, Japan)

TB3-8 *Interactive self-reflection architecture using cellular automata* ……458
K. Takadama, K. Shimohara (ATR, Japan)
K. Hajiri (Sony Computer Science Labs. Inc., Japan)

TB4: Robotics III (General Session) (Room 305)

TB4-1 *A study on modeling and simulation of human-like pose of robot arms* ……462
K. Oura (Waseda University, Japan)
C. Sato, I. Hanazaki (Tokyo Denki University, Japan)

TB4-2 *Simplification of visual sensing in skill-based manipulation* ……466
A. Nakamura, T. Suehiro (Electrotechnical Laboratory, Japan)
T. Ogasawara (Nara Institute of Science and Technology, Japan)
H. Tsukune (Oita Industrial Research Institute, Japan)

TB4-3 *SlugBot: a robotic predator in the natural world.* ……470
I. Kelly, O. Holland, C. Melhuish
(University of the West of England, UK)

TB4-4 *The predictive operation of a micro robot* ……476
M. Sugisaka, Y. Ninomiya (Oita University, Japan)

TC1: Complexity II (General Session) (Room B)

TC1-1 *Cooperative action controller of multi-agent system* ……480
Y. B. Kim, H. M. Jang, D. J. Kim, Y. K. Choi, S. Kim
(Pusan National University, Korea)

TC1-2 *Labeling Q-learning for partially observable Markov decision process environments* ……484
H. Lee, K. Abe (Tohoku University, Japan)
H. Kamaya (Hachinohe College of Technology, Japan)

TC1-3 *Multi agent simulation of Sika deers with GIS data* ……488
G. Yamamoto, K. Yamasaki, K. Hara, Y. Kinouchi
(Tokyo University of Information Sciences, Japan)

TC1-4 *Complex control approach for Sumo robot* ……491
M. Sugisaka, S. Mishiro (Oita University, Japan)

TC1-5 *Applications of conscious models to human-machine cooperative systems* ……495
K. Nakazato (Ishikawajima-Harima Heavy Industries Co., Ltd., Japan)
E. Bamba (Kinki University, Japan)

TC2: Super Mechano-Systems (General Session) (Room B)

TC2-1 *Nonlinear control of snake robot* ……499
P. Prautsch, T. Mita (Tokyo Institute of Technology, Japan)

TC2-2 *Stabilizing control of passive biped robot and its application to active walking* ……503
F. Asano, M. Yamakita, K. Furuta (Tokyo Institute of Technology, Japan)

TC2-3 *Control of a snake robot with redundancy based on kinematic model* ……507
K. Mogi, F. Matsuno (Tokyo Institute of Technology, Japan)

TC2-4 *Autonomous formation of a mechanical structure by a cellular automaton* ……511
N. Inou, K. Okumura, S. Ujihashi (Tokyo Institute of Technology, Japan)

TC2-5 *An evolutionary optimization method for structural dynamics - to optimize the structural components of super mechano-systems -* ……515
M. Okuma, K. Hashidume (Tokyo Institute of Technology, Japan)
H. Sakuma (Mitsubishi Heavy Industry, Ltd., Japan)

TC3: Neural Networks I (General Session) (Room B)

TC3-1 *A circularly linked synergetic neural networks* ……519
M. Nakagawa (Nagaoka University of Technology, Japan)

TC3-2 *Neural network human-face recognition using wavelet mosaic pattern* ……523
L. Zhang, H. Kondo (Kyushu Institute of Technology, Japan)

TC3-3 *Evolutionary neural network based on DNA coding method* ……528
W. H. Lee, H. Kang (Chung-Ang University, Korea)

TC3-4 *Application of the neural networks for the diagnosis of blood constituents* ……532
M. T. Yamawaki (Miyazaki University, Japan)
S. Yoshida (Oita Medical University, Japan)

TC3-5 *Recognition of the sequential motion of the hand by the GRBF network* ……534
M. Hirakawa (Fukuoka International University, Japan)
M. Ishibashi, T. Murakami, M. Okamoto
(Kyushu Institute of Technology, Japan)

TC3-6 *Forecasting and precision on using multi-layer neural networks: introducing of auxiliary functions* ……539
T. Teshima, H. Zhu, T. Aoyama, I. Yoshihara (Miyazaki University, Japan)

TC3-7 *Long range forecasting for chaos by using multi-layer neural networks: Ikeda's chaos* ……543
H. Zhu, T. Teshima, T. Aoyama, I. Yoshihara (Miyazaki University, Japan)

TC3-8 *Characteristics of an adaptive associative memory system based on autonomous image reaction* ……547
Y. Kinouchi, N. Matsumoto, M. Mizutani
(Tokyo University of Information Sciences, Japan)
S. Inabayashi (Pacific Systems Corporation, Japan)

TC4: Genetic Algorithms III (General Session) (Room B)

TC4-1 *Optimization of wavelet filters for image compression by using genetic algorithm* ……551
J. Tang (Beijing University of Posts and Telecommunications, China)
J. Sun (Shanghai Pudong Development Bank, China)
Y. G. Zhang (Academia Sinica, China)

TC4-2 *Genetic algorithm-like expression of reproductive and competitive RBF network* ……555
K. Okuhara, T. Tanaka (Hiroshima Prefectural University, Japan)

TC4-3 *Introducing a new genetic operator, schema extraction, to solve the traveling salesman problem* ……559
Y. G. Jo, H. Kang (Chung-Ang University, Korea)

TC4-4 *A reinforcement learning scheme based on decision tree representation of state space and its genetic acquisition* ……563
H. Fujiki, I. Ono, N. Ono (The University of Tokushima, Japan)

TC4-5 *Approximation of a nonlinear function using neural network based on genetic algorithm* ……568
M. Sugisaka, F. Dai (Oita University, Japan)

TD1: Evolutionary Computations II (General Session) (Room 302)

TD1-1 *Neural network-like hierarchical sociogenesis as a common logic underlying evolutionary semeiogenesis and machinogenesis in various levels of organic individuals* ……572
K. Ohnishi, D. Kanbe (Niigata University, Japan)

TD1-2 *Evolution of modular structures for multiple reinforcement learning agents* ……576
T. Nijo, I. Ono, N. Ono (The University of Tokushima, Japan)

TD1-3 *An evolutionary approach to the synthesis of stable grasp in multi-fingered robot hand* ……580
Y. Katada, M. Svinin, Y. Matsumura, K. Ohkura, K. Ueda
(Kobe University, Japan)

TD1-4 *Evolution of multi-agent system strategies using genetic programming* ……584
M. Sugisaka, X. Wang (Oita University, Japan)

TD1-5 *Emergence of individuality and sociality by reinforcement learning in multi-agent systems* ……589
K. Shibata, M. Ueda, K. Itou (Tokyo Institute of Technology, Japan)

TD2: Mobile Vehicles II (General Session) (Room 302)

TD2-1 *Experiments of a wheel type mobile robot* ……593
M. Sugisaka, M. Hara (Oita University, Japan)

TD2-2 *Autonomous mobile vehicle* ……597
M. Sugisaka, T. Adachi (Oita University, Japan)

TD2-3 *Studies on 3-D image processing for autonomous vehicle* ……601
M. Sugisaka, N. Mitsuo (Oita University, Japan)

TD2-4 *The steering control of the mobile vehicle* ……605
M. Sugisaka, S. Seki (Oita University, Japan)

TD3: Mobile Vehicles III & Related Fields (General Session) (Room 302)

TD3-1 *Capturing a moving object using an active camera mounted on a mobile robot* ……609
J. W. Park, J. H. Park, H. R. Hur, J. M. Lee
(Pusan National University, Korea)
K. Tagawa, H. Haneda (Kobe University, Japan)

TD3-2 *Development of mobile robot for autonomous navigation on line of welding joint by using genetic algorithm* ……613
S. Kushida, T. Haseda, D. Y. Ju (Saitama Institute of Technology, Japan)
M. Kuge (ASPECTS INC., Japan)

TD3-3 *Real-time image processing and lane recognition algorithm for autonomous driving* ……617
S. T. Park, S. Y. Yang, J. H. Jung (University of Ulsan, Korea)
J. I. Bae (Pukyong National University, Korea)
M. H. Lee (Pusan National University, Korea)

TD3-4 *The development of inverter for generation of electricity from solar panel* ……621
M. Sugisaka, K. Sakamoto (Oita University, Japan)

TD3-5 *Switching Petri Nets as a tool for user interface* ……625
Y. Funami (Shonai College of Industry and Technology, Japan)
K. Yamada (Yamagata University, Japan)

TD3-6 *Stochastic diffusion control for gene regulation protein* ……629
H. Hirayama, N. Kitagawa (Asahikawa Medical College, Japan)
Y. Okita (Shizuoka University, Japan)
T. Kazui (Hamamatsu Medical University, Japan)

TD3-7 *Extracting an independent component with a microphone array* ……633
T. Iwamoto (Mitsubishi Electric Corp., Japan)

TD3-8 *H2 control strategy for ion channels on the biological membrane* ……637
H. Hirayama, N. Kitagawa (Asahikawa Medical College, Japan)
Y. Okita (Shizuoka University, Japan)
T. Kazui (Hamamatsu Medical College, Japan)

FA1: Artificial Life II (General Session) (Room 304)

FA1-1 *From human error to "agent error"* ……641
L. Chaudron, N. Maille (Onera-Cert, France)

FA1-2 *Why people play: artificial lives acquiring play-instinct to stabilize productivity* ……643
S. Tamura (Osaka University, Japan)
T. Yokouchi, S. Inabayashi, W. Hayakawa, Y. Katou
(Pacific Systems Corporation, Japan)

FA1-3 *The study of the language occurrence model using Associatron* ……647
M. Sugisaka, H. Aito (Oita University, Japan)

FA1-4 *Artificial knowledge system constructed interconnectional network modeled on the natural brain* ……651
N. Nishinomiya, Y. Yonezawa (Ibaraki University, Japan)

FA1-5 *Multi-module reinforcement learning with situation decomposition algorithm* ……657
O. Itou, T. Omori (Tokyo University of Agriculture and Technology, Japan)
H. Okada, H. Yamakawa (Fujitsu Laboratories Ltd., Japan)

FA1-6 *Multi-agent reinforcement learning applied to a chase problem in a continuous world* ……661
H. Tamakoshi, S. Ishii (Nara Institute of Science and Technology, Japan)

FA2: Machine Intelligence and Robotic Control (Organized Session) (Room 304)

FA2-1 *Intelligent control of inverted pendulum using LQR and neural network* ……665
T. Fujinaka, S. Omatu (Osaka Prefecture University, Japan)

FA2-2 *Dynamic identification of an industrial seven-link manipulator by evolving Runge-Kutta-Gill RBF networks* ……669
T. Nanayakkara, K. Watanabe, K. Kiguchi, K. Izumi
(Saga University, Japan)

FA2-3 *Control of robot manipulators via chaotic attractors and fuzzy model-based regulators* ……673
L. Udawatta, K. Watanabe, K. Kiguchi, K. Izumi (Saga University, Japan)

FA2-4 *A dynamic control of holonomic and omnidirectional mobile robot platforms with two active dual-wheel caster assemblies* ……677
F. Han, K. Watanabe, K. Kiguchi, K. Izumi (Saga University, Japan)

FA2-5 *Environment cognition for a quadruped robot using an ART-based neural network* ……681
X. Chen, K. Watanabe, K. Kiguchi, K. Izumi (Saga University, Japan)

FB1: Biologically Inspired Systems (Organized Session) (Room 305)

FB1-1 *Symbiotic artifact design by DOILS: design organization by information in life systems* ……685
O. Katai, H. Kawakami, O. Toba (Kyoto University, Japan)

FB1-2 *Design of shared autonomy system based on biologically-inspired coordination* ……689
T. Sawaragi, Y. Horiguchi (Kyoto University, Japan)

FB1-3 *Development of virtual collaborator as an innovative interface agent system between human and plant systems: its framework, present status and future direction* ……693
H. Yoshikawa, H. Shimoda, W. Wu, H. Ishii, K. Ito (Kyoto University, Japan)

FB1-4 *A simulation model to study an adaptive behavior of fish school under the dead-locked status* ……699
H. Nakamine (Kyoto University of Education, Japan)
N. Sannomiya (Kyoto Institute of Technology, Japan)

FB1-5 *Autonomous decentralized control of a quadruped locomotion robot using oscillators* ……703
K. Tsujita, A. Onat, K. Tsuchiya, Y. Kawano (Kyoto University, Japan)

FB2: Adaptive, Learning and Distributed Control in Animals (Organized Session) (Room 305)

FB2-1 *How human learns to control a non-linear system in iterative tracking tasks?* ……711
M. Arif (PIEAS, Pakistan)
H. Inooka, T. Ishihara (Tohoku University, Japan)

FB2-2 *Emergence of optimal gait pattern of insect in the real world attaining its required direction and velocity* ……717
M. Sato, K. Akimoto, Y. Makino, M. Yano (Tohoku University, Japan)

FB2-3 *Breakthrough for artificial heart control based on adaptability of circulatory system* ……721
M. Yoshizawa, A. Tanaka, K. Abe, T. Yambe, S. Nitta
(Tohoku University, Japan)
H. Takeda (Tohoku Gakuin University, Japan)

FB2-4 *Collective behavior of multi-agent system with simple interaction* ……725
K. Sugawara, T. Watanabe
(The University of Electro-Communications, Japan)
M. Sano, K. Abe (Tohoku University, Japan)
I. Yoshihara (Miyazaki University, Japan)

FB3: Life and Computation (Organized Session) (Room 305)

FB3-1 *On the possibility of "complex systems biology"* ……728
H. Tanaka (Tokyo Medical and Dental University, Japan)

FB3-2 *A compact 8-node Linux PC cluster for protein information analysis* ……729
Y. Akiyama, T. Noguchi, K. Onizuka, M. Ando
(Real World Computing Partnership, Japan)

FB3-3 *Inference of nonlinear biological systems by using linear programming* ……733
T. Akutsu, S. Miyano (The University of Tokyo, Japan)
S. Kuhara (Kyushu University, Japan)

FB3-4 *Bio-computing in the 21st century* ……737
M. Arita (Electrotechnical Laboratory, Japan)

FB3-5 *Analysis of viral evolution pattern by using mixed evolution model* ……741
F. Ren, S. Ogishima, H. Tanaka
(Tokyo Medical and Dental University, Japan)

FB3-6 *Efficiencies of information criteria for topology selection in reconstructing molecular phylogenetic tree* ……745
S. Ogishima, F. Ren, H. Tanaka
(Tokyo Medical and Dental University, Japan)

FB3-7 *On a model of artificial cell* ……749
Y. Suzuki, H. Tanaka (Tokyo Medical and Dental University, Japan)

FC1: Neural Network Applications (Organized Session) (Room B)

FC1-1 *Propagation of stochastic signals in probabilistic universal learning networks and validation of its calculation method* ……753
C. Jin, K. Hirasawa, J. Murata, J. Hu (Kyushu University, Japan)

FC1-2 *The convergency of GA for neural network* ……757
A. Uchibori, M. Oka (Ube National College of Technology, Japan),
T. Satoh, K. Tanaka (Yamaguchi University, Japan)

FC1-3 *Independent component analysis using time delayed sampling* ……761
M. Yoshioka, S. Omatu (Osaka Prefecture University, Japan)

FC1-4 *Design and experimental evaluation of a neural-net based multivariable PID controller* ……765

T. Yamamoto (Hiroshima University, Japan)
T. Oki (Okayama Prefectural University, Japan)
S. L. Shah (University of Alberta, Canada)

FC1-5 *Position control of hard disk driver by intelligent approach* ……769

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

FC1-6 *Application of fuzzy behavior-based control for a PUMA robot* ……773

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

FC2: Genomic Dynamics and DNA Computing (Organized Session) (Room B)

FC2-1 *DNA computing --- a novel paradigm of artificial life in vitro* ……777

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

FC2-2 *DNA computing by genomic dynamics I: --- evolutionary modeling of emergence and context-sensible grammar representation* ……781

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

FC2-3 *DNA computing by genomic dynamics II: --- a simulation wetware prototype of dynamical DNA computation* ……785

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

FC2-4 *Personal adaptation of human touching behavior recognizer using pressure sensors* ……789

K. Shinozawa, F. Naya, J. Yamato
(NTT Communication Science Labs, Japan)

FD1: Robust Virtual Engineering (General Session) (Room 309)

FD1-1 *A conceptual framework on geino organization model for artificial geino performers with lives* ……793

Y. Kawamura (Osaka University of Economics and Law, Japan)
T. Ogata (Yamanashi University, Japan)

FD1-2 *Human characteristic on phase discrimination between visual and auditory perception by using two-dimensional sine wave stimuli* ……797

O. Nobuki, Y. Sakai (Yamaguchi University, Japan)
J. L. Wu (Kagawa University, Japan)

FD1-3 *Measurement of human visual and auditory calculate characteristic by functional magnetic resonance imaging (fMRI)* ……801

H. Mizuhara (Yamaguchi University, Japan)
J. L. Wu (Kagawa University, Japan)
K. Negoro, M. Hashida, J. Ogasawara, S. Yamauchi, N. Matsunaga
(Yamaguchi University School of Medicine, Japan)

FD1-4 *Subjective contours strength depends on the product of the length of the subjective edges* ……805

S. Ge, S. Kawano (Yamaguchi University, Japan)
J. L. Wu (Kagawa University, Japan)

FD1-5 *Designing a pattern generator for a walking robot by modeling the human creative process* ……809

T. Inamoto, H. Murao, V. V. Kryssanov, Y. Kurematsu, S. Kitamura
(Kobe University, Japan)

FD1-6 *Home medical support system by bilateral telecommication between patients and doctors* ……813

H. Wakamatsu, S. Honma, X. Zhang
(Tokyo Medical and Dental University, Japan)
H. Yamamoto (Ashikaga Institute of Technology, Japan)

FD2: Neurocomputing Technologies (General Session) (Room 309)

FD2-1 *Modular neural network architecture for inverse modeling of mobile robots* ……817
S. Yamaguchi, H. Itakura (Chiba Institute of Technology, Japan)

FD2-2 *Design space tradeoff in VLSI implementations of mixed-signal neuro-fuzzy processors* ……821
J. Madrenas, E. Alarcon, J. Cosp, J. M. Moreno, A. Poveda,
J. Cabestany (Technical University of Catalunya, Spain)

FD2-3 *Simulations of construction learning as for neuron-computer resources* ……825
T. Aoyama, H. Zhu, T. Teshima, I. Yoshihara (Miyazaki University, Japan)

FD2-4 *Ultra-precision position control of a piezoelectric actuator using neuro-PD controller* ……829
B. R. Lee, S. W. Jeong, K. Y. Park (University of Ulsan, Korea)

FD2-5 *Parameter identification and trajectory control of a DC motor using neural networks* ……833
D. S. Bae (Chang-Won Polytechnic College, Korea)
J. I. Bae (Pukyung National University, Korea)
J. M. Lee (Pusan National University, Korea)

FD3: Neural Networks II (General Session) (Room 309)

FD3-1 *Emergence of cognitive space in rat navigation - a neural network model of the shortcut problem* ……838
Y. Aota, Y. Miyake (Tokyo Institute of Technology, Japan)
S. Ukai (Yokohama National University, Japan)

FD3-2 *A new method to improve generalization ability of neural networks* ……842
B. Lu, K. Hirasawa, J. Murata (Kyushu University, Japan)

FD3-3 *Random iterated neural networks as dynamical systems universal approximators* ……846
F. Nino, F. Botelho, G. Hernandez, A. Quas
(University of Memphis, Colombia)

FD3-4 *Reinforcement learning with autonomous state space construction using unsupervised clustering method* ……850
K. Yamada, M. Svinin, K. Ueda (Kobe University, Japan)

FD3-5 *The study on the optimization problem by neural network* ……854
M. Sugisaka, N. Watanabe (Oita University, Japan)

FD3-6 *The prediction amount of LD gas consumption by neural network* ……858
M. Sugisaka, H. Koga (Oita University, Japan)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Interactive Self-Reflection Model: Aim and Overview

○ Keiki Takadama
ATR Human Information Processing Research Labs.
2-2 Hikaridai, Seika-cho, Soraku-gun
Kyoto 619-0288 Japan
keiki@hip.atr.co.jp
Tel: +81-774-95-1007, Fax: +81-774-95-1008

Katsunori Shimohara
ATR Human Information Processing Research Labs.
2-2 Hikaridai, Seika-cho, Soraku-gun
Kyoto 619-0288 Japan
katsu@hip.atr.co.jp
Tel: +81-774-95-1070, Fax: +81-774-95-1008

Abstract

This paper proposes a novel concept of *Interactive Self-Reflection (ISR)*, describes its aims and overview, and claims its importance and necessity to analyze properties of embedded elements that are ordinarily pre-determined by human designers. Concretely, ISR includes the following three elements that (1) precisely recognize the current situation by determining the boundary between self and others, (2) appropriately detect or produce the required information from certain information weighted according to a sense of values, and (3) autonomously generate both the goal and evaluation criteria according to the sense of values. To address ISR, this paper explores its possibilities and discusses its technical challenging issues.

Keywords: interactive, self-reflection, agent-based approach, self and others, sense of values, goal and evaluation criteria

1 Introduction

What is autonomy? How can autonomy be implemented? To answer these questions, a lot of research has tried to address especially in the context of AI (Artificial Intelligence) [6]. These attempts have led the proposal of many methods and techniques that enable systems to adapt to unknown environments. However, the adaptability in most these works is shown only in pre-determined experimental domains, and most likely do not work in other domains. This is because the following three essential elements are determined for a system by its human designer in advance: (1) the situation to be met by the system, (2) the required information, and (3) the goal of the system and evaluation criteria. These assumptions obviously prevent us from discussing autonomous systems in the true sense.

To directly address this issue for overcoming the above assumptions, we propose a novel concept called *Interactive Self-Reflection (ISR)*, which includes the following three mechanisms: (1) precise recognition of the current situation by determining the boundary between self and others, (2) appropriate detection or production of the required information from certain information weighted according to a sense of values, and (3) autonomous generation of both the goal and evaluation criteria according to the sense of values. We use the word *ISR* because the boundary between self and others dynamically changes through "interactive" actions among agents and the sense of values, goal, and evaluation criteria are self-organized or self-generated as a result of "self-reflection".

Employing this concept, this paper claims an importance and necessity of ISR to analyze properties of embedded elements that are ordinarily pre-determined by human designers. In consideration of the fact that most conventional research has concentrated on improving particular methods or techniques in their domains, ISR gives a new focus on the investigation of essential parts of autonomy. From this characteristic, ISR does not aim to narrow an argument down to methods or techniques that solve given problems effectively but aims to discuss embedded and essential elements in autonomy that derive problems themselves instead of solving problems. Although ISR does not cover all analyses of autonomy, this paper starts to address this issue by employing the three mechanisms in ISR.

This paper is organized as follows. Section 2 starts by describing the concept of ISR, and Section 3 discusses technical challenging issues for ISR. Finally, conclusions are given in Section 4.

2 Interactive Self-Reflection

2.1 Architecture of ISR

ISR is composed of many agents, and each agent in ISR has the same architecture, which includes the following problem solver and mechanisms as shown in Fig. 1. In this model, an environment includes other agents, a family and an organization which the agents belong to, outsiders, and so on. From this definition

* Paper submitted to the Fifth International Symposium on Artificial Life and Robotics (AROB'2000)

of an environment, the agents in ISR are assumed as the subject at certain times and are assumed as the environment at other times. To distinguish this difference, Fig. 1 shows the boundary line between self and others.

< **Problem Solver** >

- **Detector and Effector** translate a part of an environmental state into the internal state of an agent and derive actions based on the internal state [6], respectively.

< **Mechanisms** >

- **(1) Boundary setting, (2) sense of values, and (3) goal and evaluation criteria** are key elements of ISR (The details are described later).

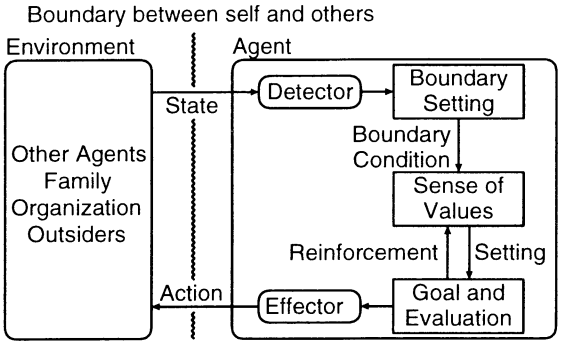


Figure 1: **Architecture of ISR**

2.2 Mechanisms in ISR

2.2.1 Boundary Setting

In ISR, the *boundary setting* mechanism precisely recognizes the current situation by determining the boundary between self and others. For example, as shown in Fig. 2, each agents decides its behavior as an individual, as a family, or as an organization according to the current situation. Each agent also decides its behavior according to the situations of others or outsiders. These situations are all indicated as one of the environmental states through the detector. From this characteristic, it is one of the key points how ISR addresses environmental states by determining several boundaries that cover the range from the individual level to the organization/outsider level.

2.2.2 Sense of Values

The *Sense of Values (SOVs)* mechanism in ISR appropriately detects or produces the required information from certain information weighted according to the SOVs. As shown in Fig. 3, ISR is composed of (1) a self part and (2) an out of self part. In detail, the former part includes (1) criteria as an individual,

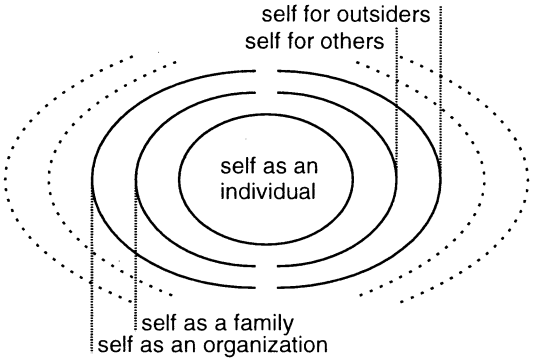


Figure 2: **Boundary Setting**

and the latter part includes (2-a) criteria as a family, (2-b) criteria as an organization, (2-c) criteria for others, and (2-d) criteria for outsiders. Note that (2-a) and (2-b) work as internal criteria, and (2-c) and (2-d) work as external criteria.

In the SOVs, each criterion has a weight which indicates its worth, and affects the information detection or production. For example, one criterion is probabilistically selected according to the weight of each criterion and ISR detects or produces information related to the selected criterion. In this case, the boundary condition decided by the boundary setting mechanism directly affects the criterion selection as shown in Fig 1. The detected or produced information includes information on individuals, others, outsiders, and so on, and affects the behavior decisions of agents. From this characteristic, it is one of the key points how ISR appropriately detects or produces the required information through the criterion selection.

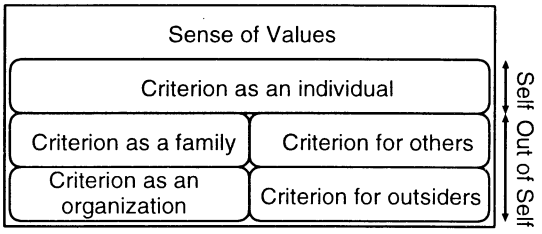


Figure 3: **Sense of Values**

2.2.3 Goal and Evaluation Criteria

Finally, the *Goal and Evaluation Criteria* mechanisms in ISR autonomously generate both the goal and evaluation criteria that leads some actions of agents through the effector. In detail, when an agent achieves the current goal, a new goal and new evaluation criteria are generated according to the SOVs (mainly criterion as an individual in the SOVs) or the current situation as shown in Fig. 1. However, it is not certain whether

agents will achieve their goals accidentally or not. It is also not certain whether both the generated goal and evaluation criteria will be appropriate or not in the current situation.*1 Due to these uncertainties, agents achieve their goals at certain times but not at other times. Therefore, agents in ISR learn to achieve their goals securely by positively reinforcing the selected criterion of the SOVs in successful cases and by negatively reinforcing them in failure cases as shown in Fig. 1. From this characteristic, it is one of the key points how ISR appropriately generates both the goal and evaluation criteria for the current situation.

2.3 Characteristics of ISR

According to Koestler, an *adaptation* is claimed as follows: "In fact, the animal does not merely adapt to the environment, but constantly adapts the environment to itself [5]." From the viewpoint of this claim, ISR has the following same aspects.

- **Adaptation to the environment:** Both a new goal and new evaluation criteria in ISR are generated whether agents achieve their goals accidentally or not. This leads to frequent changes of both goals and evaluation criteria. Although these frequent changes do not seem to be effective from the viewpoint of learning, they are quite essential elements for adaptation to the environment. In particular, it is not certain whether the generated goal and evaluation criteria will be appropriate or not for the current situation, and thus the above frequent changes offer the potential to find appropriate ones. From this characteristic, the goal and evaluation criteria mechanism in ISR enables agents to *adapt to the environment* in terms of searching for appropriate goals and evaluation criteria.
- **Adaptation from the environment:** The SOVs mechanism, on the other hand, positively and negatively reinforces criteria in the SOVs according to the results, and self-organizes the SOVs through this reinforcement. Since this self-organized SOVs enables agents to maintain their identities,*2 the agents can keep their same behaviors. From this characteristic, the SOVs mechanism in ISR *adapts the environment* (including other agents) *to the agents* in terms of understanding the agents.

From the above characteristics, the goal and evaluation criteria mechanism works as "adaptation to the environment" by frequently changing both the goals and evaluation criteria, and the SOVs mechanism works as "adaptation from the environment" by maintaining the identities of agents. This indicates that (1) ISR includes both aspects of adaptation that Koestler claims and (2) the interaction between both aspects of adaptation has a great potential for "autonomy". We, however, claim here that it is not sufficient to implement autonomy from this potential, because a small change such as the interaction among agents leads to different results especially in the complex environment. This means that it is important to decide direction for the adaptation rather than to activate the interaction of both aspects of adaptation. In ISR, the boundary setting mechanism works as the corresponded axis as shown in Fig. 4. Since our previous research has also shown that a similar integration contributes to improving the system performance [7], we claim that the integration of three elements in ISR is one way for implementing autonomy in the true sense.

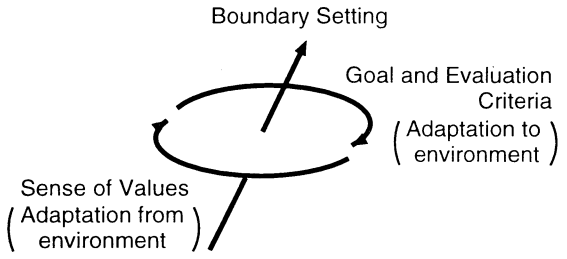


Figure 4: **Three axes in ISR**

3 Challenging Issues

3.1 Boundary Setting

In complex environments where a lot of agents interact with each other, it is difficult for agents to decide their appropriate behaviors. This is because a small change in interaction among agents leads to different results as described in the previous section. This tendency becomes conspicuous, especially when we conduct social or economic simulations to investigate effects of a prediction in the stock market or a prediction on a new organizational structure. To overcome this problem, the following issues must be addressed to precisely recognize the current situation: (1) boundary determination between self and others and (2) direction determination according to the boundary settings.

*1 For example, an agent generates its own goal (*e.g.*, to sing a song) and decides its evaluation (*e.g.*, a large voice is good) at the individual level, but the agent must behave at the organization level in some situations (*e.g.*, a movie theater).

*2 ISR assumes that identity is implemented by the SOVs. Although a lot of discussions can be made as to whether this assumption is correct or not, this paper starts by employing this assumption to analyze properties of embedded elements that are ordinarily pre-determined by human designers.

In detail, the first issue should determine the standpoint of the agents as described in section 2.2.1. Some parts of this issue are related to research in prisoner's dilemma [1], but the first issue covers a wider range of agent standpoints. The second issue should determine some directions for agents, and we must consider what kinds of directions are needed or required for the boundary setting mechanism.

3.2 Sense of Values

If the necessary information can be obtained by selecting from or combining given information, systems can solve given problems. This assumption, however, is not realistic and consequently systems based on this assumption are hard to address the unexpected situations (*e.g.*, airplane accidents) or unconsidered situations (*e.g.*, the year 2000 problem). To overcome this problem, the following issues must be addressed to appropriately detect or produce the required information from certain information: (1) information detection based on situation awareness [3], (2) information processing through a regulation mechanism, and (3) information creation for overcoming symbol grading problems. In detail, the first issue should mainly detect information by focusing on some points or parts, yet cover the area/range of the utilized information. Some parts of this issue are solved by knowledge discovery and data mining techniques [4]. The second issue should process information by regulating information from rough or frame information. One of the approaches for this is successfully employed in a robot navigation problem [2]. In ISR, required information is regulated from sense of values. Finally, the third should issue create a new or different axis of information. From the above three issues, we must investigate how to implement these mechanisms.

3.3 Goal and Evaluation Criteria

Whenever human designers setup some problems, most of the goals and evaluation criteria are tacitly decided at the same time. This makes it difficult for systems to solve various kinds of problems autonomously. Furthermore, some of the information may be omitted in the setup of the problems. Considering these facts, problem setting is not done from the viewpoint of agents but from the viewpoint of human designers, and accordingly it is not certain whether the determined goals and evaluation criteria are appropriate or not for agents. To overcome this problem, the following issues must be addressed to autonomously generate both goals and evaluation criteria for agents: (1) sub-goal generation and (2) undecidable evaluation criteria. In detail, the first issue should generate sub-goals from a large goal. This large goal can be given by human designers, but agents must divide the goal into a problem dependent part and a non-dependent part. We therefore need to investigate an importance of such division. The second issue should address evaluation criteria that are difficult to determine appropriately. Examples of such evaluation criteria include criteria for good communications between humans and agents. Since this issue is related to a lot of topics such as goal, sense of values, emotion, and so on, a careful investigation is required to analyze the relationship between evaluation criteria and others.

4 Conclusion

This paper has proposed a novel concept of *Interactive Self-Reflection (ISR)*, described its aims and overview, and claimed its importance and necessity to analyze properties of embedded elements that are ordinarily pre-determined by human designers. Furthermore, this paper has explored possibilities for ISR and discussed its technical challenging issues. Our future research will address the issues described in section 3.

Acknowledgment

The authors wish to thank Dr. T. Maeshiro from ATR Human Information Processing Research Labs., Dr. T. Yamaguchi from Nara National College of Technology, and Mr. T. Shiose from Kyoto University for their useful comments and helpful discussions.

References

[1] R.M. Axelrod: *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*, Princeton University Press, 1997.

[2] P. Eggenberger et al.: "Toward Seamless Transfer from Simulated to Real Worlds: A Dynamically-Rearranging Neural Network Approach", *The Eighth European Workshop on Learning Robot (EWLR-8)*, pp. 4–13, 1999.

[3] M.R. Endsley: "Toward a Theory of Situation Awareness in Dynamic Systems", *Human Factors*, Vol. 37, No. 1, pp. 32–64, 1995.

[4] U.M. Fayyad et al.: *Advances in Knowledge Discovery and Data Mining*, AAAI/MIT Press, 1996.

[5] A. Koestler: *The Act of Creation*, Penguin Books, 1964.

[6] S.J. Russell and P. Norving: *Artificial Intelligence: A Modern Approach*, Prentice-Hall International, 1995.

[7] K. Takadama et al.: "How to design good rules for multiple learning agents in scheduling problems?", in *The 2nd Pacific Rim International Workshop on Multi-Agents (PRIMA'99)*, H. Nakashima and C. Zhang (Eds.), Lecture Notes in Artificial Intelligence, Volume 1733, Springer-Verlag, pp. 126–140, 1999.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Learning is Equal to Participation: Interactive Acquisition of Identity in a Group

*Takayuki Shiose *Tetsuo Sawaragi **Osamu Katai +Michio Okada
*Graduate School of Eng. **Graduate School of Info. +MIC Labs.
Kyoto Univ. Kyoto Univ. ATR
Kyoto, 606-8501 Kyoto, 606-8501 Seika-cho, 619-0288
shiose/sawaragi@prec.kyoto-u.ac.jp katai@sys.i.kyoto-u.ac.jp okada@mic.atr.co.jp

Abstract

What is *autonomous* learning for artificial agents? This must be distinguished from learning for autonomous agents. The latter deals with agents' ability to acquire an action series based on one fixed perspective. On the contrary, we think that autonomous learning must deal with agents' abilities to reconstruct their own perspectives themselves against problems as a target. Conventional learning model for autonomous agents like a reinforcement learning has assumed that an agent's perspective is expressed as an evaluation function, which was mostly designed by designers in advance. In realizing an autonomous learning agent's abilities to reconstruct its own evaluation function as well as to validate that change without any supervision, the only sources an agent can refer to are their own interactions with other agents and their own history of such interaction. We regard such a duality of referencing capabilities as a minimally necessary basic architecture for autonomous learning and develop this idea as a "bi-referential model". We briefly introduce our proposing bi-referential model and discuss what factors can affect upon a variety of behaviors emerging out of interactions of multiple agents equipped with a bi-referential model. Through this behavioral analysis, we conclude that autonomous learning is equivalent to an evolutional change of the ways of agents' *participation* in a joint activity a process of which is governed apart from individual selves.

1 Introduction

In conventional studies on the design of autonomous artificial agents, agent's perspective on a criterion for behaviors has been fixed and given externally by a designer in advance. Therein, an agent never doubts or evaluate whether its own perspective is appropriate or not to its surrounded situations. As far as such a design convention concerns, a variability of the agent's behaviors is fully restricted within a designer's scope on the possible situations that an agent may be though to encounter. However, such a style of design assuming that all the possible interactions and the encountering problems can be exhaustively expected at the design time is not valid. For instance, the WWW world where the web-agent wanders continues to enlarge the world day by day, and a human user, with whom the personal agent collaborates, may be changing in his ways of interacting with the world. In such situations, agents may always have a lot of possibilities to encounter novel problems, and the conventional design principle would necessitate designers to be called for to redesign the agent each time the agent meets novel problems. Therefore, what is the most needed functions for such agents is to be able to reconstruct their own perspectives when they meet new problems. Such an ability of self-reconstructing its own perspectives would free the designers from engaging in maintaining efforts for adjusting the relations between the agent and its environment. That is, an *autonomy* in a sense of leaving the agent to adapt to the novelties will contribute to informing artificial agents with new life. Allowing agents to modify their perspectives as a basis of behaving would lead to abandoning an idea to model a *living* organism (i.e., an agent) as a simple stimulus-response system. Such a paradigm is not new but is prevailing in a field of biological and/or living systems. Moriyama et al. confirm this fact from a series of experiments on octopus's maze finding behaviors and "forward going behavior" by Armadillidium vulgar. They explicated that a simple mechanical model regarding animals as input-output machines is not valid to explain behaviors of such simple animals that have no central nervous system. Actual behaviors that cannot be explained by the model have been apt to be regarded as exceptions or probabilistic errors so far, but even those are governed by some deterministic rules like "Zipf-laws" [6] [7], and such a drastic change of behaviors do emerge being triggered by an agent's

proactive internal model reconstruction.

This style of adaptation is quite different from the existing learning model like a reinforcement learning in a sense that the responsibility of this reconstruction is left to the agent itself, rather than to the designer. In a reinforcement learning model, attaining an agent's goal was equivalent to finding a policy function relating the states and appropriate action so that it can maximize the expected payoffs determined by the evaluation function. Under this formulation of learning as an optimizing problem, many achievements of searching algorithm such as a genetic algorithm and neural networks could be used. However, the fundamental for autonomous learning is far from an optimization problem with such a well-defined formulation. Being triggered by the similar motive, currently a controversy on the reward design for reinforcement learning is often discussed. Where is, the prevailing main idea is on the ways to decompose the overall reward explicated in the evaluation function to the component ones to deal with the computational tractability, and there is no idea discussed on where the evaluation function itself comes from and how it evolves dynamically. In this paper, we focus on this problem. In the following section, we explain studies on general autonomy with respect to constructing of evaluation function, and we will give comparative investigations into them from a viewpoint of our focusing agent's autonomy. Then, we introduce our proposing algorithm called a bi-referential model that let an agent change its evaluation function through referring both to the self and the others. Especially, we investigate into the relations between learning and ways of participation in a multiagent group. Finally, we conclude by explicating what characterizes an agent's autonomous learning.

2 Related Works

2.1 Autonomous Expansion of Sub-Goals

As mentioned above, for a large scale learning problem formulated in an optimization problem, a goal decomposition into sub-goals is promising to deal with the computational tractability. In terms of evaluation functions, this corresponds to converting the overall goal into components of the evaluation function, each of which specifies the individual sub-goals. But here we encounter the same problem again on who designs this goal decomposition. The limitation incurred in design of artificial agents is a designer's difficulty in exhaustively predicting the possible behaviors, and this is also the case when we deal with the goals other than the behaviors; we have to assume the existence of preliminary knowledge needed for deductively expand a goal into sub-goals and a designer has to embed this knowledge into the definition of evaluation function. Actually, in the works of [2], [3] and [5], autonomous expansion of the goal for alleviating the designer's burden to exhaustively predefine the goal structure are attempted, but the above problem necessary for realizing an agent's autonomous learning remains to be unresolved. We have to exclude the intervention of the designer out of the expansion of the goal and to let the agent itself do this upon encountering the problems.

2.2 Reconstruction in Organization

We can find an idea to reconstruct the evaluation function in action in the study of organizational learning in organization science. Argyris[1] depicted the following two aspects as fundamental for retaining an organization; *stability* and *flexibility*. This stability and flexibility are indeed contradicting functions, but a continuing organization, just like a living biological system, is actually realizing both of them by adaptively switching distinguished two phases of stabilizing and destabilizing its internal order by changing the modes of interactions among the organization members caused by the change of the norm of the organization. Argyris proposed a *dual learning* loop consisting of two distinctive types of learning; the first one is to compare the current situation of the organization with its goal according to the norm of the organization, while the latter is to doubt its norm itself. We have to note that organizational learning does not mean the existence of *dual* actors of individuals and the organization itself. Organization is a just a collection of individuals and is not an different actor having its own norm. The norm of the organization is built up as a result of the individuals' change of values, and the behavior of the organization is defined as a function of the individuals (i.e., not independent from individuals). In spite of this fact, what the organization learns is not equal to a simple sum of what the individuals learn, but more than that can be acquired by the organization. This is the essence of organization learning, and this is realized by establishing the functional duality of organizational learning within the individual agents, and the organizational behavior emerging out of their interactions will reveal both the aspects of stability and flexibility just like a living system. This duality of learning loop of the individuals is one of requisites to embed within our bi-referential model for real autonomous learning.

3 Bi-Referential Model

3.1 Definition of Bi-Referential Model

Based on the above consideration, we develop a model that allows the agent reconstruct its own evaluation function, thus embed the duality of learning within the individual agents. We have developed such a model as a bi-referential model[8], which consists of the following two referencing capabilities as shown in Fig.1. The first one, called a *social-referential model*, is to acquire a set of the agent's action rules so that they satisfy an agent's own evaluation function as a standard for its action-execution. The second one, called a *self-referential model*, is to reconstruct its own evaluation function in doubting the proprieties of itself. We adopted the classifier system for realizing each referencing capability. We investigated into the validity of our proposing model by simulation experiments, in which group behaviors of agents, named "primitive soccer" agents, having the model were investigated. This is a "primitive" game in which each player can find out his/her "joy" (i.e., positive reward) just in engaging in the common activity and in sharing the common experiences with other colleague agents. The main characteristics of this game is that there exists no global objective and each player gets satisfied in sustaining a status of participating and engaging in some shared activity with others. Only with this assumption, each agent attempts to adaptively acquire behavioral rules by accumulating the experiences of interacting with others. Herein, no criteria for behavior formation is taught externally in a form of neither penalties nor rewards, but must be self-mediated by altering their own beliefs and senses of value by the agents themselves.

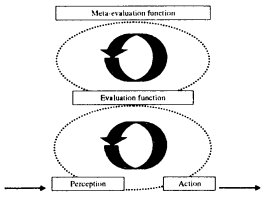


Figure 1: An Architecture of Bi-Referential Model

We aimed at the agent's autonomous reconstruction of their own evaluation function, but we relaxes this restriction by defining a number of alternatives of evaluation functions and let the agent adaptively adopt one of those. We think that this relaxation is inevitable as long as we take an approach of computer simulations. With respect to this problem, Gunji solves to make new functions recurrently by giving a "construction map" theory[4]. In our modeling, the selection of evaluation functions in the self-referential model is done depending upon temporal groups in which each agent participates. As a domain of evaluation functions we used two primitive desires of *egoism* and *altruism* after the idea of developmental psychology. The function used for the agent's evaluation is written by the following equations:

$$r = R_T(t) \equiv (W_1 * near(t) + W_2 * contact(t)) * Egoism_T + (W_3 * number(t) + W_4 * collision(t)) * Altruism_T \quad (1)$$

where the following parameters are weighing coefficients and are determined by trial and error as follows; $W_1 = 0.5, W_2 = 0.5, W_3 = 1.0, and W_4 = -0.2$. Parameters of *Egoism* and *Altruism* are balancing index denoting how the agent thinks highly of its self-doing activity and of its commitments into other agents' activities. They are changeable only via self-referential model and we call this space that is defined above these two parameters an agent's "space of value".

The social-referential model plays a role of acquiring action series that can keep the value of evaluation function defined in equation(1) as higher as possible. On the other hand, the role of the self-referential model is to adjust the two parameters of *Egosim* and *Altruism*, both of which define the evaluation function of the social-referential model in referring to the past along its own time scale, that is different from a time scale of the social-referential model. Context memory $t = C_T(i, a, r)$ memorizes the value of evaluation function at every step of past T times. The self-referential model has an objective of increasing a value of $\Phi(T)$ as follows,

$$\Phi(T) = \sum^{T} \frac{dR_T(t)}{dt}. \quad (2)$$

The conditional part of a rule in the self-referential classifier specifies current satisfying levels of the performance of the social-referencing classifier system. As values of this variable, we define three discrete values of $\Phi(T)$ by dividing its domain into three intervals (NB: Negative Big, NS: Negative Small, P: Positive). The actional part of the self-referencing classifier specifies the ways to change the value of the above mentioned two parameters; to increase u units, not to change, or to decrease u units. These adjustments correspond to changing an agent's sense of values. Different from preparing one evaluation function, this

definition of $\Phi(T)$ decides only trends of changing evaluation functions from the prepared domain of that.

3.2 Indirect Interactions

In our system, no agent can directly observe the evaluation function of other agents, since each learning mechanism is "self-closed" within the individuals. Each agent can only indirectly validate its own evaluation function by accumulating its own history of interactions with other agents and by reflecting on those. In spite of the lack of any explicit communications, agents' positions in the space of value are determined dependent upon others' positions. Moreover, we found that the mutual relations emerged among their evaluation functions are sensitively affected by the degrees of competition of shared resources. In this sense, that position is formed as a niche for a life[8]. Fig.2 shows interdependence among different agents. Wherein, we can observe that segregation of evaluation function is dependent upon the accidental nature in the early stages of interactions just like general ecological systems.

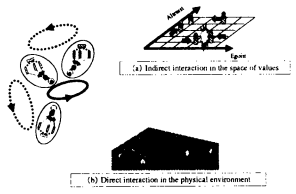


Figure 2: Indirect Interaction on the Space of Value

4 Summary

In this paper, we proposed a bi-referential model for reconsidering what an autonomous learning for agents is. We could find that each agent equipped with this model and its associated algorithm could let indirect interdependence emerge even if each learning architecture is self-closed. When some agents experience the shortage of shared resources enough to satisfy its own sense of value, some agents amplify their own interests in the relevant resources and others decline that. These processes can be explained as conflicts of values as if they play "shove and push" with other agents on the space of value. And we could find out that the processes of such segregation on the space of value are greatly influenced by the history what kinds of groups the agent participates in and what kinds of interactions the agent experiences. This is just similar to the concept of "social identity". Therefore, we cannot specify what is a valid sense of value in advance and there is not any solution for validating appropriate sense of value except for the actual participation to the relevant group.

For the agent to validate its own behavior, the agent must *ground* the evaluation of its behavior on the group which the agent participates in. In this way, the agent must repeat the above mentioned process against each behavior one by one. However, what the agent should do under the undefined environment is to reconstruct the evaluation standard by itself. In our model, a self referential capability for artificial agents is done on the basis of referring the base of expansion of the goal into sub-goals to its external environments, but it is not completed until they establish its feedback to self-reflection phase in an interactive way. In this sense, autonomous learning is strictly dependent upon the history of how the agent participates in the group as its environment.

References

[1] C. Argyris and D.A. Schön, *Organizational Learning*, Addison-Wesley, 1978.

[2] J.A.Boyan and A.W.Moore, Learning Evaluation Functions for Global Optimization and Boolean Satisfiability, *the Proceeding of AAAI-98*, pp.3-10.

[3] R.Catrambone and K.J.Holyoak, Learning subgoals and methods for solving probability problems, Memory & Cognition, Vol.18, 6, 1990.

[4] Y.Gunji, K.Ito, Y.Kusunoki, Formal model of internal measurement: Alternate changing between recursive definition and domain equation, *Physica D*, 110, pp.289-312, 1997.

[5] K. Morizane, S. Yamada, and J. Toyoda, Robot Learning for Moving between Subgoals, *the Fourth Pacific Rim International Conference on Artificial Intelligence*, 1996.

[6] T.Moriyama, Y.Gunji, Autonomous Learning in Maze Solution by Octpus, *Ethology*, 103, 1997.

[7] T.Moriyama, Anticipatory Behavior in Animals, *the Proceeding of the Second International Conference on Computing Anticipatory Systems '98*, pp.121-129, 1998.

[8] T.Shiose, T.Sawaragi, O.Katai, M.Okada, Dynamics of Reciprocal Learning by Bi-Referential Model within Multiagent Systems, *Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics*, Vol.4, pp.4045-4050, 1998.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Interactive Self-Reflection Model for Communications

Norihisa Okada[1,2,3], Katsunori Shimohara[1,2], Keiki Takadama[2], Michio Okada[1,3] and Osamu Katai[1]

[1] Dept. of Systems Science, Graduate School of Informatics, Kyoto University
Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, JAPAN

[2] ATR Human Information Processing Research Labs, Kyoto, Japan

[3] ATR Media Integration & Communications Research Labs, Kyoto, Japan
2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02, Japan

E-mail: okada@sys.i.kyoto-u.ac.jp

Abstract

This paper proposes an interactive self-reflection model for communications in order to examine mutual restrictions between superficial strategies and the sense of values in communications. This model has a bi-referential structure that consists of two learning mechanisms, i.e. 'social-reference' and 'self-reference' , which represent communication strategies and the sense of values respectively. The model is formulated on a two-person Iterated Prisoner's Dilemma game. Simulations show that the proposed model can generate diverse patterns of mutual restriction, and imply that such mutual restriction plays an essential role in communications.

Keywords: self-reference, sense of values, communication strategy, mutual restriction, Iterated Prisoner's Dilemma Game.

1. Introduction

Through communications humans are able to share something as the result of the communications. This is a general view of communications. However, are we really able to share something? Such a thought might be a sort of illusion. Even if we seem to have good communications superficially, at times we feel unsatisfied at a deep level in our minds. That is, superficial communications do not echo the sense of values in our deep inner minds.

What influence is caused on the superficial communications of communicating people by the differences in their sense of values? Or, what influence is caused on their sense of values by their superficial communications? These are the essential research issues that we are working on.

Postulating that an interface agent in a computer has a sense of values, on the other hand, what will happen in the communications between the agent and a human? We can at least expect that the agent to enrich and diversify the communications, like humans do with each other. From an engineering viewpoint, as well as cognitive viewpoint, how to achieve a such mechanism in a computer agent is also a very important and interesting research issue.

As the first step toward tackling the above research issues, in this paper, we propose an interactive self-reflection model for communications in order to examine mutual restrictions between superficial strategies and the sense of values in communications. We employ a bi-referential model [1], which consists of two sub- learning mechanisms, as the basis of the model proposed here. One of the mechanisms is defined as 'social-reference' and represents strategies used for superficial interactions between two communicating entities. The other is defined as 'self-reference' and represents the sense of values as the internal states. These mechanisms have been implemented into a two-person Iterated Prisoner's Dilemma game [2].

2. Iterated Prisoner's Dilemma

Iterated Prisoner's Dilemma (IPD) is one of the famous frameworks able to deal with psychological behaviors of the human mind in computers. In IPD, players have a payoff matrix to strategies they can take, as shown in Table.1. In this table, C indicates cooperative and D indicates defective. The reward that a player gets depends on his partner's action in this game. For example, if the player wants to keep a good relationship with his partner, the player should take C. In this case, however, if the partner takes D, the player will lose

more than he expected. If the player wants to get a rich reward soon, he should take D. In this case, however, if the partner takes D, you will again lose more than he expected.

Accordingly, the players have to face a kind of trade-off –a dilemma, that is, whether they should be cooperative or defective. Although many studies have been conducted so far to find some effective strategies and conditions, no best strategy exists.

At present, we are thinking of utilizing IPD as a framework for our research, that is, as a model of interaction between two agents. One idea is to regard the payoff matrix as the sense of values of the players, which evaluates and then affects the strategies they take for interactions. In addition, we want the payoff matrix as the sense of values to be changeable in this usage; the payoff matrix remains unchangeable in the conventional usage of IPD.

Table.1 Payoff matrix

	C	D
C	3	1
D	4	2

3. Interactive Self-Reflection Model

3.1. Architecture

As described before, we employ a "two-person Iterated Prisoner's Dilemma Game" as the framework for interactions between two agents, and implement a bi-referential model into the framework, as shown in Figure 1. Each agent has a payoff matrix as its sense of values, self-strategies for evaluating rewards, and social strategies for determining the next move. The 'social-reference' of the bi-referential model works as a learning mechanism to generate and evaluate social strategies. The 'self-reference' of the bi-referential model works as a learning mechanism to modify and update the sense of values kept with the payoff matrix.

How sensitive the learning mechanisms is one of the factors to control the overall performance of this model. Basically, however, 'social-reference' corresponds to short-term learning, and 'self-reference' to long-term learning.

3.2. Bi-reference Learning Mechanisms

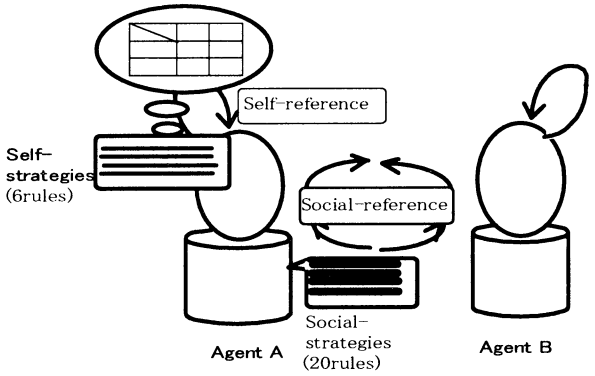


Fig.1 Bi referencial model

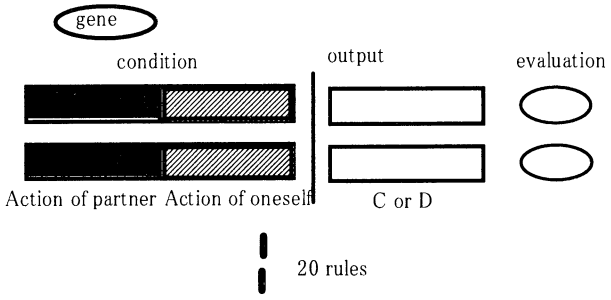


Fig.2 strategies of social-reference

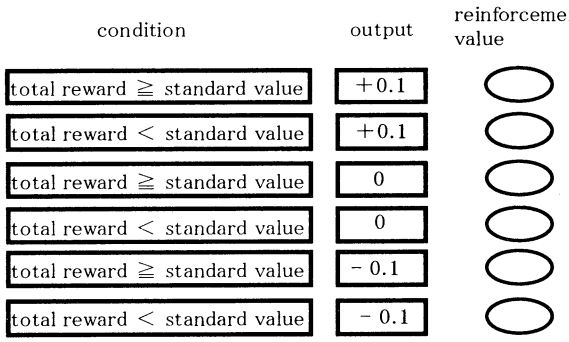


Fig. 3 Strategies of self-reference

3.2.1. Social-reference Learning Mechanism

Here, we have prepared 20 rules for social strategies, each of which consists of three action pairs of both agents in the past as a condition and the next move (C or D) as output, as shown in Figure 2. By using the three most recent action pairs as input, if the input matches the condition of a specific rule, the next move of the rule should be executed. If there are plural rules each of which condition matches the input, the next move of the rule, of which the evaluation value is the highest, should be executed. When there is no rule with a condition that matches the input, C or D should be randomly selected. In this case, the rule that has the same condition as the input and the next move as one selected should be added to the rule set of social strategies.

The 'evaluation value' of each rule is calculated by the following equation after every rule execution.

$$(evalation) = \frac{(total_reward_of_the_rule)}{(the_number_of_fire)}$$

In this equation, the total reward is calculated by using the payoff matrices of the agents, as will be described later. Consequently, all rules get 'evaluation values' at the same time after every rule execution. As we are now limiting the number of rules to 20, if the number exceeds this limit, the rule with the lowest 'evaluation value' is removed from the set.

Since there is no warranty that the mechanism mentioned above will get an appropriate strategy, we have introduced a Genetic Algorithm (GA) as well. That is, after every 20 interactions (i.e., after playing the game for 20 times), genetic operations such as selection, crossover, and mutation are independently applied to the rule set so that the agents can acquire appropriate strategies.

3.2.2. Self-reference Learning Mechanism

Assuming the values of the payoff matrix to be the sense of values of an agent, this mechanism provides the agent with a means to make some changes to the values of the matrix. Here, we have prepared six different rules for the self-strategies, as shown in Figure 3, to enable the agent to modify and update the values. Each rule has a condition part, ant output that indicates the degree of change, and a reinforcement value.

Learning is performed after every 20 x 5 = 100 interactions, as follows: Agents get a reward for every interaction, and compare total rewards with the standard value predetermined in the rules. If the condition part of a rule is satisfied, the rule should be selected. There are always three options, as you can see in Figure 3. However, one of the rules is selected based on roulette selection in terms of the reinforcement values. The rule is fired, and the values of the matrix for the sense of values are updated according to the degree of change indicated by the rule.

If the total reward that an agent gets becomes larger than the previous one, the fired rule is reinforced. In contrast, if the total reward becomes smaller, the rule is weakened.

In the following simulation, the unit update is set as 0.1 point, as shown in Figure 4. The range of values is from 1.0 to -1.0.

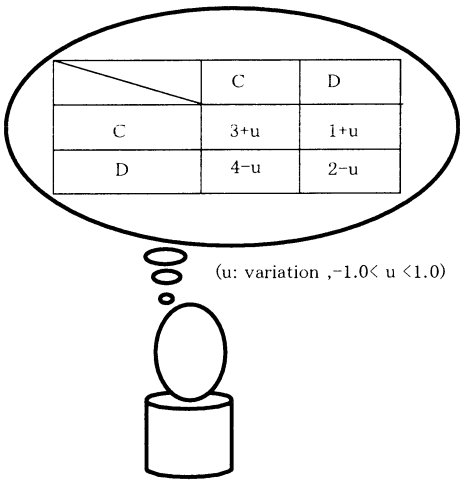


Fig.4 changing the sense of velues

3.2.3. How to relate the Two Learning Mechanisms

The two learning mechanisms in this paper are related through the total reward that an agent gets, while the total reward is the criterion of the self-reference learning mechanism. Depending on the total reward, therefore, the agent decides how to change the values of its payoff matrix. Using the changed payoff matrix, the agent calculates its reward and evaluates the rules.

In this paper, as the first step, we employ a simple way to relate the two mechanisms described above. However, how to relate the two learning mechanisms should be one of the essential factors controlling the overall performance of this model. There are various ways that can be considered, which have enough room for accepting different ideas form the further studies.

4. Simulation and Discussion

In our simulations, two agents played the Two-person IPD, and each agent had the rule set independently which included 20 rules, as mentioned above. One generation was defined as 20 x 5 interactions in the following.

Figure 5 shows the rate of cooperation versus generations, and Figure 6 shows the internal states of agents, which imply the sense of values, versus generations.

In this simulation, the two learning mechanisms were connected indirectly through the total reward values, as shown in Figure 7. In particular, this kind of mutual restriction was easily found around 300 generations as shown in Figure 6.

In addition, from these figures, we were able to summarize the interactions both superficially and in the deep mind as follows: "At first, agent A could not have good communi-

cations with agent B even superficially, and didn't want to cooperate with agent B in the sense of values. As their interactions went on, however, agent A became aware of agent B's cooperative behavior, and decided to cooperate with agent B. After a while, agent A seemed to have some doubts about agent B. Eventually, however, agent A came to believe agent B and recovered their cooperative relationship."

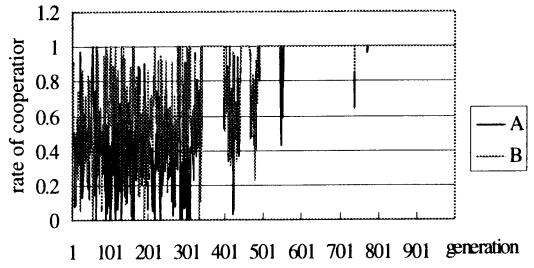


Fig.5 The rate of cooperation in each generation

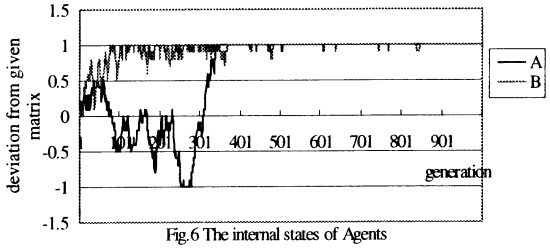


Fig.6 The internal states of Agents

We could observe a phenomenon where the completely opposite interpretation was made. The proposed model is not for the emergence of cooperative and/or defective relationships between agents. In that sense, therefore, we could summarize that the proposed self-reflection model can provide a rich structure to generate various and diverse interactions, depending on the mutual restrictions between superficial strategies and the sense of values in communications.

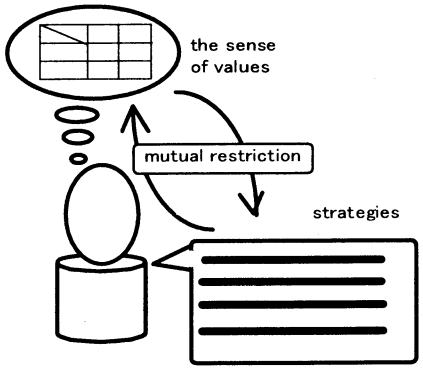


Fig.7 mutual restriction

5. Conclusion

We have proposed an interactive self-reflection model for communications in order to examine mutual restrictions between superficial strategies and the sense of values in communications. The model has been implemented as a bi-referential structure that consists of two learning mechanisms, i.e., 'social-reference' and 'self-reference', which represent communication strategies and the sense of values respectively. We have formulated the model using the framework of a two-person Iterated Prisoner's Dilemma game and conducted simulations on the interactions between two agents. Through the simulations, we could ascertain that the proposed model can generate diverse patterns of mutual restrictions, and that such mutual restrictions play an essential role in communications.

The future direction of this study will be in applying this model to situations where the number of agents is more than two, and investigating how the sense of values influences the formation of groups. Creating an agent who will be able to find its criterion for changing the sense of values through interactions with others, and controlling communication phenomena by means of the self-reference learning mechanism will also be challenged in the future.

References

[1] T.Shiose, T.Sawaragi, O.Katai, &M.Okada (1998). "Dynamics of Teciprocal Learning by Bi-Referencial Model within Multiagent Systems". *Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics*, 4, 4045-4050.

[2] R.Axelrod(1984). *The Evolution of Cooperation* Basic Book Inc.

[3] A.Ito(1996). "How Do Selfish Agents Learn to Cooperate?", *Artificial Life V*, Langton and Shimohara, pp.183-192.

[4]T.Ishida,K.Suzuki,Y.Kakazu(1996), "Dynamic Payoff Function Based on karma Converge on Iterated Prisoner's Dilemma", *Proceedings of the 10th Annual Conference of JSAI* pp.597-600 (in Japanese).

[5] M.Taiji, T.Ikegami(1999), " Dynamics of Learning Agents in Dilemma Games " , *Cognitive Studies*, March 1999, pp.21-30 (in Japanese).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Interactive Self-Reflection based Reinforcement Learning for Multiagent Coordination

Tomohiro YAMAGUCHI (yamaguch@info.mara-k.ac.jp)
Department of Information Engineering, Nara National College of Technology
22 Yata-cyo, Yamato-koriyama city, Nara, Japan

Abstract

One of the main goals of Artificial Intelligence is to realizing an intelligent agent that behaves autonomously by its sense of value. Reinforcement learning (RL, in short) is the major learning mechanism for the agent to adapt itself to various situations of an unknown environment flexibly. The merit of RL is that the design cost is small. Only giving the agent a goal state as setting a reward, a set of optimal behavior sequences toward the goal state from each state can be learned by trial and error. However, in a multiagent system (MAS, in short) environment that has mutual dependency among agents, it is difficult for a human to setup suitable learning goals for each agent, besides the existent framework of RL that aims for objective and egoistic optimality is inadequate. Therefore, it requires the active and interactive learning function that treats how to coordinate the interaction among other learning agents. This paper presents a new framework of reinforcement learning to generate and coordinate the learning goals interactively among agents. Then coordinating conflict of interests among the agents is discussed to dissolve the social dilemma.

Keywords: reinforcement learning, multiagent, self-reflection, reinforcement signal, communication

1. Introduction

One of the main goals of Artificial Intelligence is to realize an intelligent agent that behaves autonomously by its sense of value. **Reinforcement learning** (**RL**, in short) is the major learning mechanism for the agent to adapt itself to various situations of an unknown environment flexibly. The merit of RL is that the design cost is small. Only giving the agent a goal state as setting a reward, a set of optimal behavior sequences toward the goal state from each state called the optimal policy can be learned by trial and error. While for a complicated problem, it takes much time to learn the optimal policy.

Since the learning goal is given by a human designer, most previous researches have not been discussed how the agent designs its learning goal autonomously. However, to shift the viewpoint of learning from the human designer to the agent, two self-reflection problems of learning occur as follows:

1) the autonomous generation of the learning goal,
2) to decompose a complicated goal or task into appropriate subgoals that guide the learning, namely how to divide into independent sub-problems.

These are important issues because it controls not only the learning efficiency but also the adaptability to a change in the environment.

A **multiagent system** (**MAS**, in short) is an exemplary dynamic environment for a learning agent since it has mutual dependency among agents. Then it is difficult for a human to setup suitable learning goals for each agent, besides the existent framework of RL that aims for objective and egoistic optimality is inadequate in the MAS environment. Therefore, it requires the active and interactive learning function that treats how to coordinate the interaction among other learning agents. The reason is that the basic theory of RL for each agent is to maximize the individual profit as the reward acquisition efficiency under a static environment. However, a social dilemma that the sum of the maximized individual profits does not maximize the group profit occurs when the MAS has the mutual dependency among agents. The cause of this is called the credit-assignment problem to appropriately evaluate the contribution of the individual behaviors in the group.

Yet, without dissolving the mutual dependency, huge computation costs prevent us from applying the RL method that explores an optimal policy in the MAS environment. One of the promising solutions of this problem is to coordinate the goals of each agent dynamically to make the sum of the maximized individual profits into maximizing the group profit, i.e. to reduce the social dilemma approximately as if it is decomposed into independent RL sub-problems of each agent.

Therefore, this paper presents a new framework of reinforcement learning to generate and coordinate the learning goals interactively among agents. Then coordinating conflict of interests among the agents is discussed to dissolve the social dilemma. In this paper, the new framework of RL is described by the following three steps:

1) Active generation of a learning goal by self-generating a reinforcement signal.
2) Active reinforcement learning by selecting a reinforcement signal.
3) Interactive self-reflection of learning goals by communicating reinforcement signals.

The point of this research is to shift the learning goal of an agent from a reward as the exceptional information to a reinforcement signal as one of the signals in the environment. A reinforcement signal is a part of the information that the agent senses or outputs in the environment. As each agent autonomously selects its reinforcement signal for the learning from its sensory input, it is possible to make its own learning goal by itself. In addition, constraining and guiding other agent's behavior learning is available to output the part of its internal states as a reinforcement signal to the environment.

2. Background

This chapter starts with the basic framework of RL, then the problems of applying previous RL methods to MAS environment.

2.1 The Basic Framework of Reinforcement Learning

This section first describes the basic framework of RL, then discusses the difference between two major RL methods.

The basic framework of RL is an **agent** that interacts with an environment through its inputs and an output. The agent selects and executes an **action** according to the sensory inputs from the environment called the **state**. The result of the action is returned from the environment as the change of the state of the agent, and the agent receives an **reward** if the agent goes through the transition with the reward. A pair of a state and an action is called a **behavior** (simply called a **rule**), a set of the behavior to be selected for each state is called a **policy**. The objective of RL is to maximize the expected acquired reward per step, then a policy that gives the action that maximizes it of each state is called the **optimal policy**.

Since the expected acquired reward is depends on the policy, the RL agent searches the policy through the interaction with the environment. How to interpret this interaction with the environment, this paper classifies the RL researches as **the objective observation type RL** or **the subjective experience type RL**.

The observation of the former is to model the environment after the agent is out of the environment. The characteristic is to divide the learning process into the environment modeling that is independent of the policy and the optimal policy search on the model. But to model an environment, it is necessary to assume the class of the environment. The merit of the objective observation RL is to reach for the optimality, though the class of the environment is restricted to the **Markov Decision Process** (**MDP**, in short) model or the similar ones.

On the other hand, the experience of the latter is the learning result that is dependent on the behaviors or the policy. The model-less RL methods as Bucket Brigade Algorithm, Profit Sharing, and Q-Learning are the subjective experience type RL. Most of them are not converged to the optimal policy since the learning results are affected by the order or the distribution of the acquired experiences. Rather the important point of the subjective experience type RL is that it is difficult to model the environment by the objective observation since the interaction by the RL agent change the environment.

2.2 The Problems of MDP based Reinforcement Learning

This section first discusses the characteristics of the MDP based RL that is the representative method of the objective observation type RL and the problems to apply it to the MAS, then describes the subjective experience type RL as the solution method.

The objective observation type RL is realized by the MDP based RL. The characteristics are to assume the objectiveness and the completeness of the observation. However, the problems of the MDP based RL are that it is difficult to apply it to the multiagent RL in principle. The important point of the MAS is that the multiagent are inconsistent with the objective observation type RL.

Then, as learning from the agent viewpoint, the characteristics of the subjective experience type RL is to assume the subjectiveness and the imcompleteness of the observation. In this paper, an experience is defined as the subjective observation result of the interaction with the environment that depends on the agent behavior and that is from the agent viewpoint. Therefore, in the MAS environment, the subjective experience type RL is necessary because it is difficult to satisfy the assumptions of the objective and the complete observation that the MDP based RL requires.

2.3 The Issues of Multiagent Reinforcement Learning

This section describes the difficulties and the issues of the multiagent RL. The difficulties of the multiagent RL are the problem of the nondeterministics of the environment caused by the existence of the learning agents, and the learning problem by the subjective experiences that are dependent of the behavior selection by the learning agent. These two problems can not be treated as the independent problems by the agent viewpoint in the MAS. Thus, MAS seems to be non-MDP environment by each RL agent. In other words, it is difficult to define the optimality of learning because of the mutual dependencies among the learning agents.

Next, two problems of MAS by the game theoretical analysis are described. First is the profit division problem among agents. It is called the inter-agent **Credit Assignment Problem** (**CAP** in short) that is how to make a division of the acquired reward through the accomplishment of the group goal. Since the definition of the objective rationality for the CAP is difficult, it is necessary to coordinate each interest and profit assignment or to form the sense of value. Second is the problem that the sum of each individual interests of a group does not agree with the group interest. It is called a **social dilemma**. In other words, each RL agent optimizes its learning goal as pursuing his self-interest may results in the loss of the group interest of MAS.

It is necessary to setup the rewards as the learning goal in order not to conflict maximizing each individual interests with maximizing the group interest. However, it is not easy to design the rewards without the social dilemma. To solve the inter-agent CAP or the social dilemma, next chapter discusses the novel mechanism for coordinating the reinforcement signal by the communication among agents.

3. Interactive Self-Reflection based Reinforcement Learning

This chapter first reconsiders the role of the reinforcement signal in RL, then presents the new framework for the multiagent RL based on the reinforcement signal communication.

3.1 The Origins and the Role of a Reinforcement Signal and a Reward

RL is one of the machine learning method using the feedback information of learning called a **reinforcement signal** (**RS**, in short). The original meaning of reinforcement comes from the 1910's behaviorism psychology that tries to understand the mechanism of an animal behavior as the response against the stimulus. In that, reinforcement is that an animal gradually becomes to repeat a specific response with a specific stimulus such as food and so on. Contrast to it, reinforcement learning in AI is to learn the mapping from inputs to outputs for acquiring the action output that results in a reinforcement signal as much as possible.

In general, there are two types of RS, one is positive signal called **reward** that encourages the behavior, the other is a negative signal called **punishment** that restrains the behavior. The learning proceeds to reinforce the behaviors that get the reward and that avoid the punishment. So when setup a reward to a goal state, the optimal policy to reach the goal state is acquired by RL.

In behaviorism psychology area, it has been studied what is the reinforcement stimulus in the environment that controls the learner's behavior. Contrast to this, previous RL research in AI did not discuss the basic problem of what is a reward. Yet to shift the thinking from reward to RS, it is available for the RL agent to use a part of information in the environment as a RS to lead the learning.

3.2 Why Self-generating Reinforcement Signals ?

This section discusses the two issues about the autonomy for directing the learning. The first issue is the self-evaluation problem of the learning criterion that justifies extrinsic rewards in RL. What is a reward for a RL agent? A reward is the fixed goal that guides the agent's learning, is out of the agent and is given by a human designer. The reason is that if the learning agent can operate its learning criterion by itself, it may change the learning goal to justify the learning result. Therefore, it has been thought that the learning goal must be fixed and out of the agent [3].

However, in the psychology area, a novel hypothesis that extrinsic reward make the children's intrinsic interest go down is presented [4-6], and is supported experimentally [7]. Note that the intrinsic interest means a motivation of a human and is the autonomous leaning goal that is found or made by the human learner. The second issue is the problem of self-generating the learning goal of RL. If the hypothesis in the psychology area is applied to the agent learning, previous RL methods with extrinsic fixed rewards is a negative factor of autonomy of RL. Besides, it is an essential problem that there is no theory to design suitable rewards. So the learning efficiency of a complex learning goal becomes worse since the designer nor the agent itself can not setup the suitable rewards.

Therefore, to solve the multiagent learning issues discussed in chapter 2, first it is important to realize the mechanism for setup the learning goal autonomously by the learning agent itself. Moreover, a modification mechanism on the learning goals by mutual evaluation among RL agents as an idea to avoid the self-evaluation problem of the learning criterion. To integrate both ideas, a new framework for multiagent RL based on RS communication.

3.3 Communicating Reinforcement Signals

In this section, the new framework of RL based on RS communication is described by the following three steps:

1) Active generation of a learning goal by self-generating a RS.

A RL agent has a reward generation function (payoff function) to generate a RS, it is used as the reinforcement for the behavior of either itself or other's. For all rules of each state, the reward generation function that decides whether it gives the reward or not is represented by a table, it is called a **reward table** in this paper. The RL agent searches for its learning goal by setting up and modifying a value of reward of each rule in the reward table, and reinforces its executed rule by the reward setup of the reward table.

2) Active reinforcement learning by selecting a RS.

When the RL agent has a learning criterion as the reward table inside, next problem is how to setup the reward that describes, guides and directs the learning goal. If there is a fixed extrinsic reward as a group goal of multiagent, a sub-reward that is not conflict with it can be self-generated. If not, it makes the RL agent decide the learning goal on the reward table by giving the selecting criterion of RS in the environment. The candidates of the RS are the RS that is generated by others, and the co-occurred or related sensory input with the extrinsic reward. This paper uses the method that the RL agents generate the RS by the episodes that get the reward or not.

3) Interactive self-reflection of learning goals by communicating RSs.

When a RL agent executes a behavior, each RL agent in the environment generates a reward for the behavior by its reward table, then send it to the executed agent as the RS. Note that this paper distinguishes the RS generated by the executed RL agent is called **the self-reinforcement signal** (**self-RS**) from the RS sent by the other RL agent is called **the other's reinforcement**

signal (other's-RS).

After receiving the other's-RS, the executed RL agent generates Rw(i) by coordinating the other's-RS and self-RS with the following reward coordination rule (1),

$$Rw(i) = G_Rw + \alpha * S_RS + \beta * \sum O_RS \quad \cdots (1)$$

where G_Rw is a global and fixed extrinsic reward, S_RS is a self-RS, O_RS is other's-RS, $0 \leqq \alpha \leqq 1$ is the credit for self reinforcement, and $0 \leqq \beta \leqq 1$ is the credit for other's reinforcement. Using the Rw(i) of the rule (1), the RL agent updates its learning table, then the RL agent coordinates the balance between self-reinforcement and the other's reinforcement by modifying the parameter of α and β in the rule (1) that is the reinforcement criterion of itself to dissolve the mutual dependency problem.

3.4 An Illustrated Example

A simple learning experiment of competitive reward setup with two RL agents is shown as the illustrated example of presented method in section 3.3. Note that the parameter of α and β is 0.5 and is fixed. Figure 1 shows the probabilistic environment with two rewards of Rw1 and Rw2 for the experiment. There are two RL agents in the environment, a trial is that the two agents starts at state 0 until both end at state0. While they behave in the environment, each reward can be acquired by only the first found agent, if both find the same reward at once, nothing to by received. Each agent generates the self-RS and other's-RS from the received reward (0 if nothing to be received), then sends other's-RS. Each agent reinforces its learning table with rule (1), and selects its action probabilistically according to the reinforcement value of the rule. It means that each agent reinforces the behavior set of itself and other's positively if it receives the reward, or negatively if it does not received the reward.

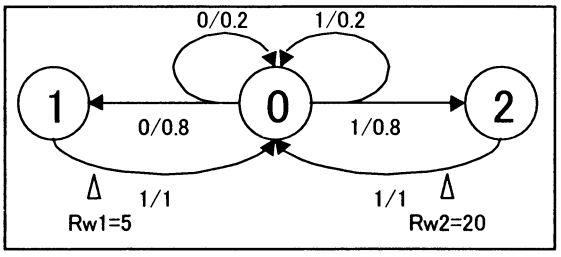


Figure 1 RL environment with two rewards

Comparative RL experiments whether other's-RS is used or not is performed. Figure 2 shows the Experimental results of the cumulative sum of the acquired rewards of two RL agents toward 100 trials. In the case of without other's-RS, the sum of the acquired reward is smaller since both agents tend to reinforce the behavior for getting Rw2. In the case of with other's-RS, the sum of the acquired reward is large because each reinforced behavior tends to be divided into different rewards.

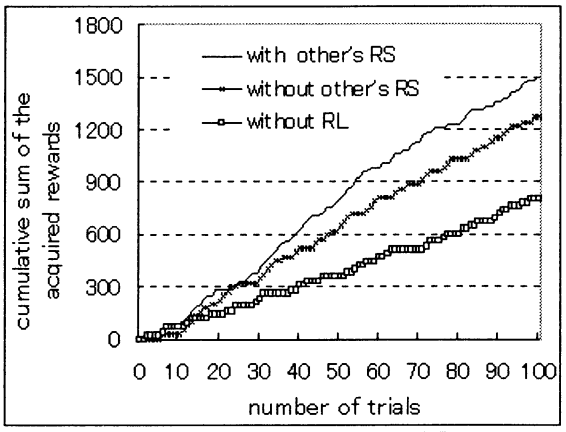


Figure 2 Experimental results of two RL agents

4. Conclusion

This paper presented the new framework of reinforcement learning to generate and coordinate the learning goals interactively among agents. Then illustrated example of the coordination of interests between two RL agents in the competitive reward is shown. The advantage of this method is that each RL agent in a group can improve their reward setup by coordinating their interests among RL agents to learn if the initial reward setup is not suitable. The limitation is that there is no idea how to build a suitable learning criterion if there is no absolute extrinsic reward. This method is applicable to an improvement of MAS performance broadly. Future work is to realize the autonomous generation of the self-learning goal for each RL agent to dissolve the social dilemma.

References

[1] Barto, A.G., Bradtke, S.J. and Singh, S.P.(1995), Learning to act using real-time dynamic programming, Artificial Intelligence 73: 81-138

[2] Sen, S. and Weiss, G. (1999), Learning in Multiagent Systems, in Multiagent Systems - A Modern Approach to Distributed Artificial Intelligence, G.Weiss (Ed.), The MIT Press, 259-298

Russell, S. and Norvig, P. (1995), Artificial Intelligence - a modern approach -, Prentice-Hall International, Inc., New Jersey, 525-529

[4] Deci, E.L. and Flaste, R. (1995), Why We do What We do - The dynamics of personal autonomy -, G.P.Putnam's Sons

[5] Deci, E.L. (1971), Effects of externally mediated rewards on intrinsic motivation, J. of Personality and Social Psychology, 18: 105-115

[6] Deci, E.L. (1972), Intrinsic motivation extrinsic reinforcement, and inequity, J. of Personality and Social Psychology, 22: 113-120

[7] Lepper, M., Greene, D. and Nisbett, R. (1973), Understanding children's intrinsic interest with extrinsic reward: A test of "overjustification" hypothesis, J. of Personality and Social Psychology, 28: 129-137

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

On the efficiency of a decentralized continuous double auction market of locally aware agents - bridging the gap between economics and interactive self-reflection

Norberto Eiji Nawa[†‡] Keiki Takadama[†] Katsunori Shimohara[†‡] Osamu Katai[‡]
[†] Dept. 6, ATR Human Information Processing Research Laboratories, Seika-cho, Kyoto 619-0288, Japan
[‡]Graduate School of Informatics, Kyoto University, Yoshida-Honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Abstract

This paper investigates a decentralized continuous double auction (CDA) market formed by traders with local information and neighborhood. A CDA is constituted by traders interested in buying and selling a given commodity. The traders interact by announcing their aimed prices to each other and performing transactions when preferences match. In this work, locality is introduced by positioning traders in a toroidal grid, and by only allowing transactions to be performed between traders within a neighborhood. Price information, thus, is only exchanged directly at the local level. Locality restricts the traders access to information on the transaction prices, which is a constraint commonly observed in ordinary markets, but overlooked in experimental studies of CDA markets. The effect of locality in the overall efficiency of the system is quantitatively analyzed through computer simulation. It is argued that the issues raised by the *interactive self-reflection model* are common to economic agents, and thus, should be of essential importance in the construction of agent-based socioeconomic models.

1 Introduction

Economic systems fit naturally in the agent-based modeling framework: they are decentralized systems, populated with heterogeneous, self-motivated, autonomous individual entities, whose interactions endogenously generate interesting phenomena in the systemic level. Despite the apparent clear resemblance between economic systems and the agent-based modeling paradigm, mainstream economics has been largely dominated by analytical, mathematically tractable models. In recent years, though, technological advances have allowed the massive use of computers for modeling economic systems, fact that may gradually change the *status quo* in economics. Research with this spirit is currently grouped under the label of *agent-based computational economics (ACE)* [11], which is inspired by the constructive methodology typically present in the studies of complex adaptive systems to the analysis and modeling of socioeconomic systems, exemplified in [2, 3, 7].

Another aspect of interest related to the study of agent-based modeling of economic systems is concerned with new insights to be acquired about novel social and economic dynamics that may emerge over the omnipresent Internet. Considering that one possible scenario for the future pictures a large population of software agents cohabiting and interacting with humans, it is of great interest to understand and analyze these possibilities in advance.

Following this rationale, this work investigates a decentralized continuous double auction (CDA) market where the traders (buyers and sellers) are subject to locality, i.e. information exchange is allowed only between neighbors, and potential partners for a transaction are restricted to those within the neighborhood of a given trader. This situation is faced by economic agents in real markets: there is no entity with access to all available information, controlling the market dynamics and flow, like an auctioneer. Contrarily, the overall dynamics is endogenously generated by the agents interactions, ultimately humans, with heterogeneous beliefs and desires, subjects to psychological factors, constantly updating their internal states according to what they perceive from their "observable environment". From this point of view, it is noticeable that the way economic agents manage this situation resembles in several aspects the issues presented by the interactive self-reflection model [10].

Despite the highly decentralized characteristic in terms of information distribution, economic systems are largely acknowledged to be perfect examples of good resource allocation and conflict resolution involving a large number of agents. Being so, it is necessary to clarify the influence of locality in the so-called market efficiency, by introducing it in the model and analyzing eventual implications.

2 Continuous Double Auction (CDA)

CDA markets, also known as open-outcry markets, have earned much interest both in theoretical and experimental economics due to their high efficiency in resource allocation. Smith's seminal work in experimental economics with a double auction scenario [9] demonstrated that even with a small number of human subjects performing the role of traders, and despite the imperfect information on the demand and supply configuration of the market, high levels of allocation efficiency were achieved. In such a scenario, the market law of demand and supply is most clear: there will be more buyers interested in purchasing a good for cheap prices, though few sellers will be willing to make deals at these same prices, and vice versa. Nevertheless, between these two ex-

tremes, there will be a price value (or an interval) where the demand of the buyers meets the supply of the sellers, which is called theoretical equilibrium price (TEP).

In the economics jargon, it is said that the TEP is the price where the market *clears*: there is no demand or supply in excess; once it is achieved, there is no pressure from the market to move the price downwards or upwards. If the internal information of each trader is clearly known (i.e., the budget of each buyer, and the lowest price each seller still can make a transaction), it is possible to determine the demand and supply curves of the market, which represent the relation between the quantity demanded and available in the market at each price (Fig. 1), and consequently determine the TEP.

The interesting results verified by Smith motivated several other works, the most remarkable ones being by Gode and Sunder [7], where by means of computer simulations it was shown that CDA markets would achieve high levels of efficiency even if the agents had "zero-intelligence", thus concluding that efficiency was due to structural properties of the CDA, and by Cliff [4], which refuted the generality of the conclusion by Gode and Sunder.

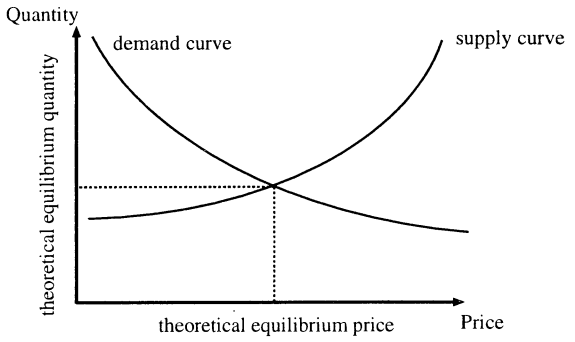


Figure 1: Demand/supply curves and the theoretical equilibrium price.

3 Local Information and Neighborhood

The main issue investigated in this paper is the effect of traders with limited, local information, in the overall efficiency of a CDA market. The basic market framework is inspired by previous computational studies of CDA, mainly [7, 4]. Locality is obtained by attributing geographical situation to the traders. By positioning them in a toroidal grid and only allowing transactions and exchange of price information between neighbors, the market is completely decentralized. As the traders are spatially distributed, the concept of neighborhood naturally arises. Although geographical information was largely neglected in past research in economics, it has been pointed out that actually it may be an aspect with several important implications in the social sciences ([6], p.12), being a central issue in works like [8, 1, 5].

Transactions are completed through direct interaction between traders, which happens as follows. Initially, the traders are given a fixed number of transaction entitlements, to be used in a "trading day". Buyers (sellers) who have valid entitlements can then publicly announce their prices, which can be only heard by those within the neighborhood of the shouting trader. Those that "heard" the shout and can afford the offer, send a counteroffer to the shouting trader. Then, the shouting trader chooses among the received counteroffers the most suitable one, and a transaction is completed.

Economic agents, in general, are very much aligned with the central issues of the interactive self-reflection model [10]. They acquire information from the environment while interacting with it, and based on this accumulated knowledge, determine their new actions. Moreover, oneself's beliefs and objectives are dynamically modified through the interaction with other elements in the system. A minimal example of such model can be pointed out in the scenario of a CDA of traders with local information and neighborhood. Traders are basically trying to solve a conflict between (1) making a deal, and (2) making a profit out from the deal. For example, if the trader is a buyer, and the aimed profit is too high, i.e. the buyer aspires a commodity for a very cheap price, it is most likely that she will face difficulties to find a suitable partner and make a deal. Thus, traders have to collect information from others, in order to adjust his own expectations on the profit that should be aimed, and increase the chances that his shout or counteroffer is accepted by someone.

4 Experiments

The computational model used the Swarm Simulation System libraries for multi-agent simulation[1]. Buyers and sellers were arranged in a checkerboard pattern in a 12x12 toroidal grid (Fig. 2), and their function, either as a buyer or as a seller, was fixed throughout the simulation. Each buyer was initially assigned a *budget* value, which represents the maximum price that she can spend in a single transaction. Accordingly, each seller was assigned a *minimum price*, which represents the lowest price that he can afford in a single transaction. These values were randomly chosen from $[2.0 \cdots 5.0]$.

The buyer's profit is the difference between her budget and the actual transaction price. Thus, when calling for potential sellers, the buyer will "shout" a price which is equal or lower than her budget; eventually there will be a seller around willing to transact on that price, and the buyer will accomplish a deal. Sellers do the opposite, inflating their shout prices over the minimum price. The seller's profit is the difference between the actual transaction price and the minimum price. Each buyer/seller works with his own *margin*, which determines the amount

[1]The code is available upon request to `xnawa@hip.atr.co.jp`. Detailed information on the Swarm Simulation System can be found at `http://www.santafe.edu/projects/swarm/`

the shout price will be lower/greater than the budget/minimum price.

When an affordable shout is heard by a trader, he sends a counteroffer. The buyer/seller counteroffer is the price he would have shouted in that moment, if his own price is greater/lower than the received shout, or the received price otherwise. The reason for trying to make a counteroffer which is better than shouted price, from the shouting trader point of view, is to increase the chances to get the transaction. Consequently, all the counteroffers that are eventually sent to the shouting trader necessarily satisfice him; thus, among the received counteroffers, the shouting trader has to choose the one that gives back the best profit, refusing the other counteroffers. This process is iterated until the "trading day" finishes, thus when the next "trading day" starts, the traders are refilled with transactions entitlements.

The traders architecture is inspired in the model presented in [4], where a set of heuristic rules determine how the traders adjust their margin values. The learning heuristic is as follows. If an affordable shout is heard by a trader τ, he makes a counteroffer, and increases his margin (i.e., buyers decrease their shout price; sellers increase them). If the heard shout is beyond τ's capability, the number of unaffordable shouts heard by τ, U_τ, is incremented. If U_τ becomes greater than the threshold T_{U_τ}, then the margin is decreased (i.e., buyers increase their shout prices, sellers decrease them) and $U_\tau \leftarrow 0$; similarly, if the number of refused counteroffers (R_τ) becomes greater than the threshold T_{R_τ}, the margin is decreased and $R_\tau \leftarrow 0$. A refused counteroffer signifies that another trader took the deal with a better (or equivalent) counteroffer. The T_{U_τ}, T_{R_τ} values of each trader are randomly chosen from $[1 \cdots 10]$.

The margin is modified by a delta-rule learning algorithm, similar to the one described in [4]. When the traders increase their margins, the aimed value is the price heard from the shouting trader plus a δa, which means that the trader always aim for an offer which is slightly better than the last affordable offer he heard; when decreasing the margins, the aimed value is his own current shout price plus/minus a D_τ percent if a buyer/seller. The initial margins, and D_τ were randomly chosen from $[5\% \cdots 35\%]$ from $[1\% \cdots 10\%]$, respectively.

The experiments consisted of running simulation sessions for a number of trading days, using different (Moore) neighborhood sizes, i.e. radius values of $r = 1, 3, 6$. Particularly for this configuration, $r = 6$ is equivalent to the situation of complete information, where every trader can hear the shouts from any other trader. In every session, all the traders had a constant and uniform value of neighborhood size.

4.1 Results

Two criteria were used to evaluate the results. The first criterion verified the *coefficient of convergence* (a), proposed in [9], and defined as $a = 100 \times \sigma_0 / P_0$, where σ_0 is the standard deviation of the actual transaction prices in a day, and P_0 is the TEP. As all the traders have the same number of entitlements and the budgets/minimum prices are samples from an uniform random distribution between 2.0 and 5.0, initially the TEP should roughly be around 3.5. Usually in economics the TEP is considered to be a static value; it is assumed that changes in the individual demands and supplies, due to trading, have little effect in the overall market. However, as the simulated market is relatively small, the TEP was recalculated at each time step, making $P_0(k)$, where k is the set of instants of time when a transaction is completed in a day.

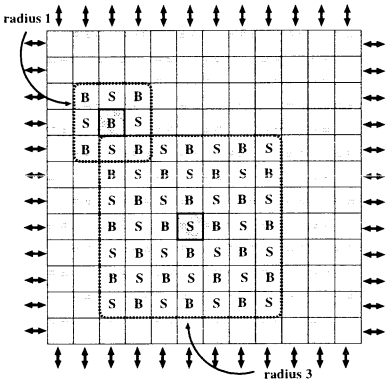


Figure 2: Toroidal grid with buyers (B) and sellers (S), radius 1 and 3 and respective neighborhoods.

The coefficient of convergence for different radius do not present significant differences, as the tradings develop (Fig. 3). As noticed in [9] with humans subjects, and [4] with computational agents, the learning capability of the traders carries the coefficient to levels close to zero, which means that the prices get very close to the TEP. Apparently, the traders neighborhood size affects very little the convergence of the market. In other words, the traders inability to know all the transaction prices until a given moment does not configure an obstacle to achieve the equilibrium price. This may be due to the fact that, although direct exchange of price information between traders occurs only within a neighborhood, in fact, as the traders adapt their shout prices, even traders in different neighborhoods can influence each other. The price information percolates through the toroidal grid. It is interesting to notice that the TEP itself is a measure of efficiency that assumes that the demand/supply *available* is also *feasible*. In the situation of constrained neighborhood and partial information ($r = 1, 3$) there is a subtle difference between both, as the demand/supply may be available in the market, but not feasible for the traders who are not within the neighborhood of the offering trader. Despite this constraint, though, the market converges to the TEP.

The effect on wealth distribution under different neighborhood sizes was also investigated. The

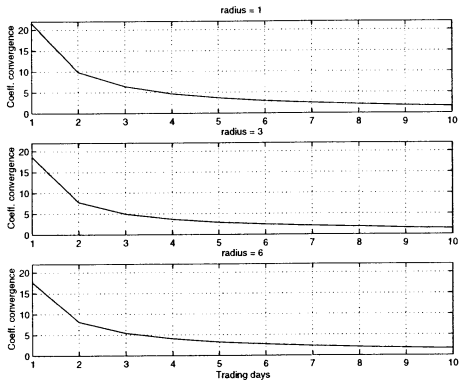


Figure 3: Coefficient of convergence for $r = 1, 3, 6$ for 10 trading days, mean values of 50 runs each.

wealth of the traders is the total sum of profits in the end of the trading days. The Hoover (H) and Gini (G) coefficients were used in order to measure the inequality in the wealth distribution in the population. The Hoover coefficient indicates the ratio of total wealth that has to be reallocated in order to reach complete equal distribution. Thus, the closer to 0, the better. The Gini coefficient is a measure of wealth concentration, where 0 represents equal distribution and 1 means that a single group in the population owns the entire wealth. The coefficients are defined as $H = \sum_{i=1}^{N}(|\frac{E_i}{E_T} - \frac{A_i}{A_T}|/2)$, and $G = 1 - \sum_{i=1}^{N} \frac{(2\times\sum_{k=1}^{i}(E_k)-E_i)\times A_i}{E_T \times A_T}$, $\frac{E_i}{A_i} > \frac{E_{i-1}}{A_{i-1}}$, where N is the number of groups in the population, E_i is the total wealth owned by the group, E_T is the total wealth in the population, A_i is the number of individuals in the group, and A_T is the total number of individuals in the population. Though not an usual procedure in the social sciences, each individual in the lattice was considered to be a single group. The results obtained are presented in Table 1. Both coefficients show that there is a slight improvement in the profit distribution when only local trading is allowed. This may be due to the fact that constraining the neighborhood, the number of potential competitors may decrease proportionally, thus, diminishing the chances that a deal is "stolen" by another trader, at least in an initial stage. These coefficients, however, do not convey much information on the geographical clustering of wealth, which is another issue of interest.

Table 1: Hoover and Gini coefficients for $r = 1, 3, 6$ for 10 trading days, mean values of 50 runs each.

radius	Hoover	Gini
1	50.6%	0.635
3	52.7%	0.643
6	53.7%	0.659

5 Conclusions

A computational model of a decentralized continuous double auction market formed by traders with local information and neighborhood was presented, and some preliminary results on the convergence to the theoretical equilibrium price and wealth distribution with different radius sizes were shown.

The results showed that, in this fairly simple framework, the convergence to the TEP occurs in both cases, when the traders are equipped with partial information/constrained neighborhood and complete information/complete neighborhood. This is an interesting result *per se*, as it confirms, at least partially, the neoclassical economics claim of market efficiency with models that are free from the complete information assumption. An interesting extension would be to invest in more sophisticated agent behaviors. In this case, all the agents had a basic *well-behaved* characteristic engraved in the learning algorithm, adjusting the internal variables in order to increase the chances of making a successful deal. It is still necessary to investigate the influence in the system of spurious agents with less predictable, tame behaviors.

Acknowledgments

Thanks to Takao Terano for suggesting the Gini coefficient. The first author receives partial financial support from CNPq, the Brazilian agency for the development of science and technology, under grant number 200050/99-0.

References

[1] P. S. Albin. *Barriers and Bounds to Rationality - Essays on Economic Complexity and Dynamics in Interactive Systems.* Princeton University Press, 1998.

[2] W. B. Arthur, J. H. Holland, B. LeBaron, R. Palmer, and P. Tayler. Asset pricing under endogenous expectations in an artificial stock market. In W. B. Arthur, S. N. Durlauf, and D. A. Lane, editors, *The Economy as an Evolving Complex System II*, pages 15–44. Addison-Wesley, 1997.

[3] N. Basu and R. J. Pryor. Growing a market economy. Technical Report SAND-97-2093, Sandia National Laboratories, September 1997. At `ftp://ftp.cs.sandia.gov/pub/rjpryor/SAND97-2093.pdf`.

[4] D. Cliff. Minimal-intelligence agents for bargaining behaviors in market-based environments. Technical Report HPL-97-91, Hewlett-Packard Labs Bristol, 1997. At `http://www.hpl.hp.com/techreports/97/HPL-97-91.html`.

[5] J. M. Epstein and R. Axtell. *Growing Artificial Societies - Social Science from the Bottom Up.* MIT Press, 1996.

[6] N. Gilbert and K. G. Troitzsch. *Simulation for the Social Scientist.* Open University Press, 1999.

[7] D. Gode and S. Sunder. Allocative efficiency of markets with zero intelligence traders: Market as a partial substitute for individual rationality. *Journal of Political Economy*, 101(1):119–137, February 1993.

[8] P. Krugman. *The Self-Organizing Economy.* Blackwell Publishers, 1996.

[9] V. L. Smith. An experimental study of competitive market behavior. *The Journal of Political Economy*, 70(2):111–137, April 1962.

[10] K. Takadama and K. Shimohara. Interactive self-reflection model: Aim and overview. In *Proc. 5^{th} Int. Symp. on Artificial Life and Robotics (AROB 5th'00)*, January 2000.

[11] L. Tesfatsion. Web site for agent based computational economics (ACE). At `http://www.econ.iastate.edu/tesfatsi/ace.htm`.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Interactive Self-Reflection Mechanism based on Homogenized Bipartite Model

Tetsuya Maeshiro
Evolutionary Systems Dept.
ATR HIP Res. Labs.
Kyoto, JAPAN 619-0288

Takayuki Morimoto
Dept. Information Science
Kanagawa University
Kanagawa, JAPAN

Yuzuru Fujiwara
Dept. Information Science
Kanagawa University
Kanagawa, JAPAN

Abstract

The Homogenized Bipartite Model (HBM) is based on the analysis of properties of information, and its representation power of semantic relationships between concepts is higher than any conventional database and knowledge base models. HBM provides multiple interpretations for the same set of concepts, suitable to realize interactive self-reflection mechanism.

1 Introduction

The design of a system with the interactive self-reflection mechanism requires an appropriate information or a concept organization/representation model, because the system must possess the external world's model. The representation model of concepts is positioned on the bottom layer of the self-reflection mechanism, and the model constitutes the base for detailed design issues such as the dynamic boundary definition, inference and learning mechanism.

The Homogenized Bipartite Model (HBM) [4] is suitable to represent semantic relationships among concepts. Another characteristic of HBM is the support for the open world interpretation, contrary to conventional models that support only the closed world interpretation.

From the graph theoretical point of view, all conventional knowledge-base and database models are either a set or a graph model. Representatives of conventional models are: entity-relation (ER) model [7], object-oriented model [7], relational model [2, 3], frame model [7], and semantic network [7]. Among these, the relational model is unique for treating the problem of representation of semantic relationship among concepts.

Section 2 lists the properties of information, Section 3 introduces HBM, and the last Section discusses and concludes.

2 Properties of Information

To build a model to represent information or concepts, it is necessary to analyze the properties of the target, which is information in this case. The following is the properties of information which all models of information should support. All conventional models, however, fail to represent these properties.

(i) Multiple hierarchical levels A concept is increasingly generalized or specialized. Recursive structure is also included.

(ii) Partial overlap Multiple concepts share common properties. This implies that concepts are not independent, and they are uncountable.

(iii) Differential description Related to (i) and (ii), where concepts are described by differences from closely related terms.

(iv) Nested structure Attributes of a concept are expanded by addition of lower level properties, caused by generation and more detailed specifications by the increase in knowledge. The nested structure is distinguished form the recursive structure, where the reference is to the self.

(v) N-ary relationships Common in real world, although all conventional models can only represent binary relations. The n-ary relation is not a combination of binary relations, as an attempt to split an n-ary relation often results in semantically meaningless relations.

(vi) Diversity of description, representation, and classification Synonyms are typical examples. To represent a protein, for instance, sequence, function and structure are possible, among others. The

number of possible representations and classifications is the number of possible combinations of attributes, resulting in a lattice structure.

(vii) Modality Multi-valued logic is necessary to represent possibilities such as *may* or *may not*, differing from two valued logic where only True and False are used. However, no numerical values commonly denoted as *confidence values* should be used, as such values depends on the viewpoint, among other reasons.

(viii) Uncountability Due to the multiple hierarchical levels and partial overlap. A concept does not exist independently from other concepts.

(ix) Relativity and duality A relationship between concepts is treated as a concept, and attributes A_i of a concept C is often treated as concepts A_i with an attribute C, depending on the viewpoint.

(x) Dynamism Description and representation of concepts depend on time and viewpoint, among others.

3 Homogenized Bipartite Model

HBM is an extension of the bipartite representation of hypergraph [1]. The hypergraph is defined by

$$E_i \neq \emptyset \tag{1}$$

$$E \subseteq 2^V \tag{2}$$

where E and V are sets of edges and vertices, respectively. These sets are not necessarily finite. The formula (2) defines edge as an n-ary relation of vertices, or the subset of the set V.

The incidence matrix of a hypergraph indicates the subsets of vertices that belong to each edge, denoted $H = (V, E)$, is a matrix of size $|V| \times |E|$, where $|V|$ and $|E|$ denote the size of the sets V and E.

The hypergraph offers duality of vertices and edges, i.e., the roles of edges and vertices are reciprocal. Let $H = (V, E)$ be a hypergraph. The dual of H, denoted H^*, is defined as $H^* = (E, V)$. Since H^* is the transpose matrix of H,

$$(H^*)^* = H \tag{3}$$

The following formulae extend the structure of hypergraph.

$$V = V \cup E \tag{4}$$

$$E = E \cup V \tag{5}$$

where the formula (4) defines recursive structures, and the formula (5) defines nested structures, homogenizing the roles of vertices V (concepts) and edges E (relationships). The formula (5) requires special attention, as the set is no longer well founded.

Finally, the formula (6) defines the conceptual structure, which is represented by a bipartite.

$$\sigma : L \rightarrow E \cup V \tag{6}$$

3.1 Representation of Concepts

Similar to the graph, the hypergraph can be represented by a bipartite, where both vertices and edges are represented by vertices (Figure 1).

HBM is represented with a graph (Figure 2), which is a bipartite representation of the extended hypergraph. A vertex represents a concept, and two or more concepts are linked if some semantic relation exists between them. From the relativity of concepts, multiple interpretations or viewpoints are possible for the same graph [5]. Defining the roles of concepts defines the viewpoint, and vice-versa. Under a viewpoint, vertices labeled with alphabets function as concepts, and numbered vertices as attributes. On the other hand, interpretation of numbered vertices as concepts is also possible. Furthermore, a vertex functions as a concept, a relationship, or an attribute, depending on the viewpoint. For instance, the vertex B is a concept with four attributes 1, 2, 3 and 4. Another possibility is the vertex B as a relationship between four concepts 1, 2, 3 and 4.

Figure 3(B) shows the representation format of a concept in HBM (Figure 3(A)). All concepts connected to the concept C are denoted as the properties, or as the property concepts of the concept. C has n property concepts, $V_1, \ldots, V_n$. Multiple activation rules or patterns are defined for each concept, and the activated rule is selected from the input pattern of roles to the concept C. The selection of activated pattern depends on the context in which the information is interpreted.

HBM has three integrity constraints. (1) Existential integrity, which constrains that every concept has a registry number and a registered name; (2) Referential integrity, which constrains that a referenced registry number and/or registered name are either an existing one or null; and (3) Reachability integrity, which constrains that no isolated concept exists, assuring the reachability of concepts by navigation.

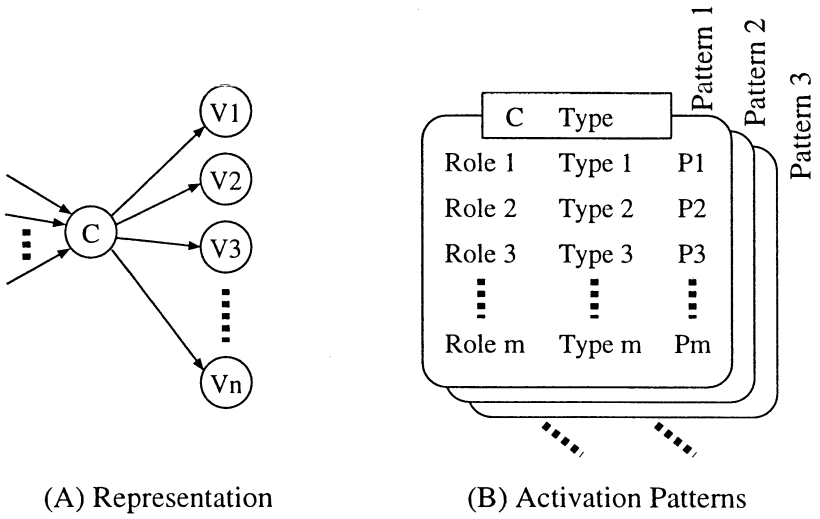


Figure 3: Representation format of concepts. "Type" is the type of the concept C. "Type" and "Type$_i$" is a concept, an attribute, or a relationship. $m \leq n$, and P_i, $1 \leq i \leq m$, represents one of $V_1 \ldots V_n$. Each V_j is specified at most once. Multiple interpretation of the concept is possible, three of which are shown (Patterns 1, 2 and 3).

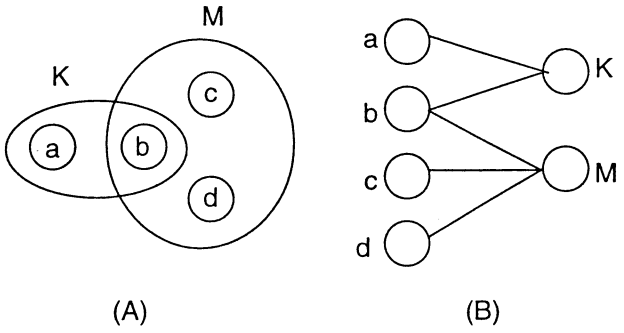


Figure 1: (A) A hypergraph and (B) its bipartite representation. In (A), a, b, c, and d are vertices, and K and M are edges.

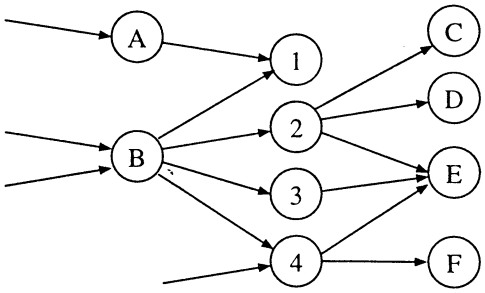


Figure 2: Concept structure represented in HBM.

3.2 Semantic Relationships

Many types of semantic relationships exist between concepts. Some examples of relationships are causal, hierarchy, association, time sequence, and possession. What determines the type of relationship is the "Role" in Figure 3, and the type of the concept C is a relationship. For instance, suppose the vertex "4" in Figure 2 represents a causal relationship between concepts E and F. Then the Type of "4" is the relationship, and the Role of E is *cause* and F is *result*. The Type of E and F can be either the concept or the relationship.

3.3 System State

The state of the system is defined as the activation pattern of concepts. The number of activation patterns is given by the number of possible roles. For a concept with n properties and r roles, there are $(r+1)^{n!}$ possibilities.

The activation sequence starts with the query input to the system, and the concept structure that matches the query is activated first. Note that the match occurs with one of interpretations of the concept. Then, according to the activation pattern of the matched rule, the concepts are repeatedly activated. Different acti-

vation pattern of concepts corresponds to a different internal state, related with the interpretation of the system state, for instance a different boundary.

4 Discussions and Conclusions

To implement HBM, a massively parallel architecture is necessary because the operation on structured information is NP-complete, for instance the concept structure search.

HBM is an information representation model that supports the open world interpretation. However, information acquisition and concept creation mechanisms are not treated in this paper, and assumes the existence of such mechanism. Further work is necessary on this topic.

Another important topic is the extraction of semantic relationships among concepts. For technical terms, which represents the concepts, it is possible to extract automatically the semantic relationships [6]. Extraction from general concepts is more difficult because there are no well defined rules like formation rules of technical terms. Related to the information acquisition mechanism mentioned above, there is a need of further study on the extraction of relationships between concepts from action and response from the environment.

The multiple viewpoints to interprete the same concept structure, offered by HBM, is related to multiple *interpreted* states of the system, where the difference among the states is due to different activation patterns of the same concept structure. And this interpretation process is another important research topic.

Acknowledgements

Authors would like to thank K. Shimohara for support.

References

[1] C. Berge, *Hypergraphs*, North-Holland, 1989.

[2] E.F. Codd, "A relational model of data for large shared data banks", *Communications of the ACM*, Vol. 13, pp.377–387, 1970.

[3] E.F. Codd, "Extending the database relational model to capture more meaning, *ACM Trans. on Database Systems*, Vol. 4, pp.397-434, 1979.

[4] Y. Fujiwara and Y. Liu, "The homogenized bipartite model for self organization of knowledge and information", *Int. Forum Inf. and Docum.*, Vol. 22, pp. 13–17, 1998.

[5] Y. Fujiwara and H. Gotoda, "Representation model for relativity of concepts", *Int. Forum Inf. and Docum.*, Vol. 20, 1995.

[6] T. Morimoto, T. Maeshiro and Y. Fujiwara, "Extraction of semantic relationships among terms to construct organized knowledge resources", *Proceedings of the First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, pp.459–465, 1999.

[7] P.H. Winston, *Artificial Intelligence*, Addison-Wesley, 1992.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Agent Architecture based on Self-Reflection Learning Classifier System

Hiroyasu Inoue*†‡ Keiki Takadama† Katsunori Shimohara*† Michio Okada*‡ Osamu Katai*

*Kyoto University Graduate School of Informatics Yoshida Honmachi, Sakyo-ku, Kyoto 606-8501 Japan
†ATR Human Information Processing Research Laboratories 2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288 Japan
‡ATR Media Integration & Communications Research Laboratories 2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288 Japan
E-mail: h-inoue@sys.i.kyoto-u.ac.jp

Abstract

This paper proposes the new agent architecture that enables an agent not only to acquire a user model through a sequence of interactions but also to adapt to changes in the user's internal states. The agent architecture is implemented by the Self-Reflection Learning Classifier System (SRLCS), which uses *exploration* and *exploitation* as its learning mechanisms to infer the user's internal states. It is shown that SRLCS has great potential for learning through two simulations in which a user model consists of either two or three different internal states.

Keywords: self-reflection, agent architecture, user modeling, internal states, learning classifier system

1 Introduction

Along with the rapid spread of computers as ubiquitous tools in our lives, the significance of agents that act as an interface between a user and a computer has been and will continue to be rapidly increasing. Some of the key issues for such interface agents include how we can enrich interactions between a user and an agent as well as seeking new possibilities of communications between them. One idea for that purpose would be to realize an agent that could give a user the sense that the agent might understand the user. As a first step toward this goal, we are attempting to create an agent that can acquire a user model and adapt to changes in the user's internal states.

How can an agent infer the user's internal states (emotions, feelings, etc.) and acquire a user model? For the first problem, many studies have already been conducted; for example, emotional information extraction from voices[5]. However those systems have the following drawbacks: (1) they are too sensitive to single actions that a user may make unintentionally; (2) they can't distinguish the same actions that a user takes in different internal states (for example, a porker-faced user). For the second problem, several studies have been conducted, too[4][7]. But they cannot adapt to the user's internal states, let alone the changes in the user's internal states.

Based on the above understanding, we propose a new agent architecture that will enable an agent not only to acquire a user model through a sequence of interactions but also to adapt to changes in the user's internal states. Emphasis is put on a mechanism to utilize a sequence of interactions between a user and an agent instead of using one specific action the user makes. As another important point, moreover, the agent architecture should have a mechanism to reuse existing knowledge about the user's internal states already acquired by the above mechanism. Here, we define the former mechanism as *exploration* and the latter mechanism as *exploitation*. Strategies of how to properly use these two mechanisms are vital to this agent architecture, and in this sense we could call it a self-reflection architecture.

The agent architecture proposed in this paper is implemented by the Self-Reflection Learning Classifier System (SRLCS), which effectively uses *exploration* and *exploitation* as its learning mechanisms to infer the user's internal states and to adapt to changes in these states.

2 Description of SRLCS

The first version of SRLCS is described here as extended LCS (Learning Classifier System). SRLCS differs from the conventional LCS in its use of different CSs (Classifier Systems) to enable an agent to adapt to changes in the user's internal states.

2.1 Conventional LCS

LCS is a general learning system in which a set of condition-action rules called classifiers compete to control a target system, gaining credit based on reinforcement from the environment[6]. As the basis of SRLCS, we employ ZCS (Zeroth level Classifier System)[6], which has no message list.

ZCS is reported to have many advantages; however, it cannot infer a user's internal state because it cannot recognize any change in the environment or in the user's internal states. As another option, there is an extended ZCS that has some memory[1]. The extended ZCS has the same problem solving ability as Holland's LCS. In the original paper, the

memory of the extended ZCS was used to recognize the environmental aliasing state that returns the same sensory configuration. Now, if we use a memory to keep the user's internal states reasoned by an agent, the extended ZCS intuitively seems capable of adapting to the user. However, even if the extended ZCS can keep the user's internal states, reasoning itself still depends on the user's only one action. Accordingly, having a memory is not essential to reasoning the user's internal states in this sense.

2.2 Architecture

To overcome such drawbacks of the conventional LCS, we propose SRLCS. Figure 1 shows the components of SRLCS, which is composed of many CSs, for an agent. These CSs have a common architecture, which includes the following problem solver, memory, and mechanisms.

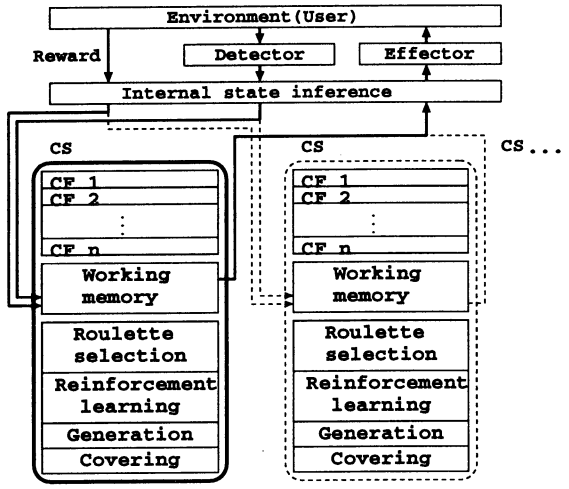


Figure 1: Components of SRLCS

< Problem Solver >

- **Detector** translates an environment state into an understandable state for the agent.
- **Effector** derives actions based on an understandable state for the agent.

< Memory >

- **CFs (Classifiers)** store a set of if-then rules with strength factors.
- **Working memory** stores recognition results of the environmental states, an agent action generated by fired rules, and reinforcement from the environment.

< Mechanisms >

- **Roulette selection** probabilistically selects one rule from the rules that match the environment. This selection depend on the strength of each classifier.
- **Reinforcement learning** strengthens classifiers to acquire a good sequence of actions. The algorithm is based on Holland's bucket brigade[3].
- **Generation** invokes the GA operation (crossover and mutation) for creating new classifiers.
- **Covering** creates the classifier when there is no classifier matching the environment.

2.3 Reasoning Mechanism

Figure 2 shows the reasoning mechanism of SRLCS, which consists of two phases: **Learning** and **Inference**. Although SRLCS can have many CSs at the same time, at a given time SRLCS executes only one CS. In the **Learning** phase shown in Fig.2, for example, SRLCS has two CSs and is executing CS_1.

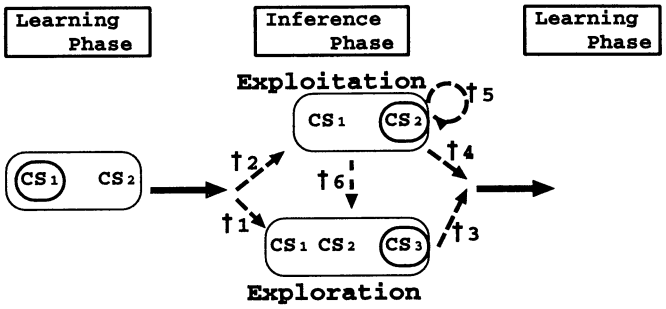


Figure 2: Reasoning Mechanism of SRLCS

The reasoning mechanism of SRLCS will be explained by using Fig.2, and in the rest of this section "†*number*" shows the number of arrows in Fig.2.

Learning: CS_1 learns as long as a user's internal state doesn't change. When the user's internal state changes, SRLCS can detect a decrease in the receiving reward by the following two ways: (a) by the convergence of the learning, or (b) by the sequence of actions that can earn the reward. For simplicity here, we take the former way as the first step in this paper.

To detect the convergence of the learning is performed as follows: SRLCS records the number of the reward during C interactions as one cycle and keeps it for R cycles[1]. Here, we use the term "interaction" to refer to a pair of user and agent actions. If |(the total of the reward in a cycle)−(the reward average of R cycles)| $< D$, SRLCS recognizes the convergence of the learning.

Once SRLCS detects the convergence of the learning, SRLCS is able to determine the decrease in receiving reward. If (the total of the reward in a cycle)$< \alpha \times$(the convergence strength), SRLCS executes the **Inference** phase, which selects **Exploration** †1 or **Exploitation** †2. SRLCS can employ many selection strategies for using **Exploration** or **Exploitation**. Although the performance depends

[1] $R \times C$ interactions

on the strategy, SRLCS here applies **Exploitation** for all of the sets of the classifiers first. If all of these trials fail, it starts **Exploration**. **Exploration** and **Exploitation** are defined as follow.

Exploration: When SRLCS infers that the user's internal state is unknown, SRLCS creates a new set of classifiers (CS_3), and starts **Learning** again †3.

Exploitation: When SRLCS infers that the user's internal state is known, SRLCS reuses the set of the classifiers (CS_2). After reusing the set of classifiers, if (the total of the reward in a cycle)> $\beta\times$(the convergence strength which was recorded in the set of classifiers), SRLCS judges its success and starts **Learning** again †4 for adapting more precisely. If not, SRLCS decides to start **Exploitation** †5 or **Exploration** †6 again. This selection depends on the strategy, as mentioned above.

3 Simulation

3.1 Simulation Framework

We have conducted simulations on the interactions between an agent with SRLCS and a postulated user model. The simulation framework we employed is shown in Fig.3. Actions from the user model to the agent are represented by 4 bits, and those from the agent to the user model by 2 bits. The agent and the user model generate an action by taking turns in their interactions.

The user model is implemented by finite state machines (FSMs), each of which corresponds to one of the user's internal states and consists of several momentary states. One of the FSMs is selected to represent the user at a given time, and a change from one FSM to another represents a change in the user's internal states. In particular, a momentary state corresponds to one user action, and this indicates that the user has no aliasing.

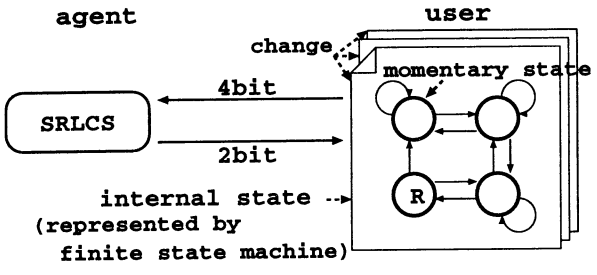


Figure 3: Overview of Simulation

3.2 Case 1: Two Internal States Problem

The first simulation was conducted for the case where the user model has two different FSMs, each of which has four momentary states. Two FSMs were switched alternatively every 2000 interactions. SRLCS only has two sets of classifiers, and it uses the strategy to exploit them alternatively. Figure 4 shows the result. The vertical axis shows the rewards SRLCS received, and the horizontal one shows the interactions plotted by every C interactions.

This result shows that ZCS's learning collapses at every change in the user's internal state. SRLCS's learning, to the contrary, collapses just temporally after the first change in the user's internal state. This shows that SRLCS can choose a set of classifiers according to the user's internal state.

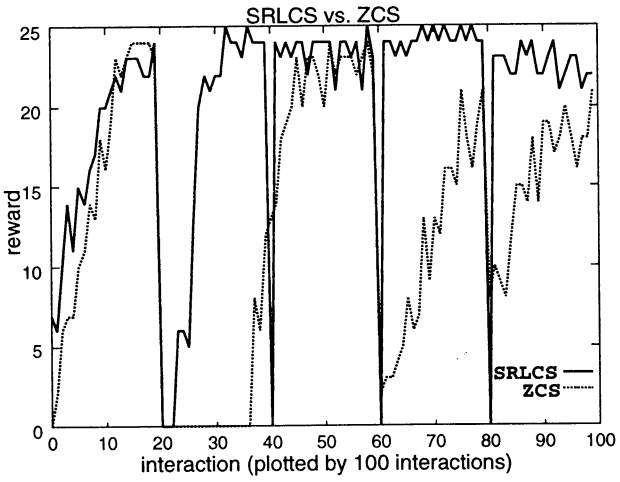


Figure 4: Result of two internal states problem. Parameters: R=5, C=100, D=3.0, α=0.5, β is not used

3.3 Case 2: Three Internal States Problem

The second simulation was conducted for the case where the user model has three different FSMs, each of which has four momentary states. Three FSMs were switched in turn as (1, 2, 3, 1, 2, 3, ...) every 3000 interactions. SRLCS initially has only one set of classifiers and uses the strategy to first exploit all of the sets of classifiers in the oldest order and then to explore if all trials fail.

Figure 5 shows the result. The meanings of the vertical axis and the horizontal one are the same as in Case 1.

This result shows that SRLCS can distinguish whether the user's internal state is known or not, and if the user's internal state is known, SRLCS can apply the correct set of classifiers. This can be understood from the result in the following manner.

Interaction 0: The user's current state is internal state 1. SRLCS learns this state.

Interaction 3000: The user's current state changes to internal state 2. SRLCS detects the change in the internal state, but there is no set of classifiers to exploit. Thus SRLCS creates a new set of classifiers to explore.

Interaction 6000: The user's current state changes to internal state 3. SRLCS detects the change and uses the set of classifiers learned for internal state 1. However, this trial fails, and thus SRLCS creates a new set of classifiers to explore.

Interaction 9000: The user's current state changes to internal state 1. SRLCS detects the change, uses the set of classifiers learned for internal state 1, and succeeds.

Interaction 12000: The user's current state changes to internal state 2. SRLCS detects the change and uses the set of classifiers learned for internal state 1. However, this trial fails. Next, SRLCS uses the set of classifiers learned for internal state 2 and succeeds.

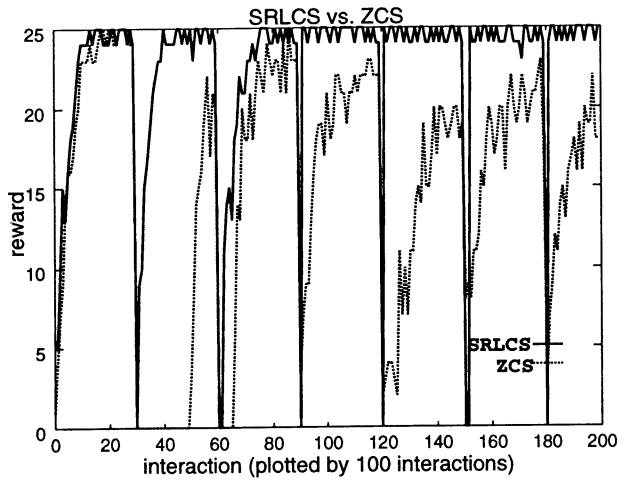


Figure 5: Result of three internal states problem. Parameters: R=10, C=100, D=2.0, α=0.5, β=0.7

4 Discussion

(1) How to Detect the Decrease in Receiving Reward: SRLCS uses the convergence of the learning to detect the decrease in receiving reward in this paper. However, this mechanism must at least know how many steps it takes to get to the convergence. That is, an agent must approximately know the number of momentary states. If SRLCS stores the sequence of actions that can earn the reward, SRLCS can make use of this sequence for detecting the decrease in receiving reward. This is because if the user's internal state changes, the agent cannot receive the reward by using the same sequence. This mechanism should be better in the sense that an agent does not need to know the number of momentary states.

(2) How to Choose Exploration or Exploitation: SRLCS's performance depends on the strategy of choosing exploration or exploitation, and various strategies can be applied as mentioned above. Strictly speaking, however, if the user does not take the same internal state previously taken, exploitation would be pointless because the set of classifiers learned would be not effective. A human being, however, probably has finite internal states and patterns of transitions between them. This implies the possibility of implementing some suitable strategy. Whether this strategy should be learned or prepared and the kind of learning method that should be used if it could be learned remain as issues for further studies.

5 Conclusions

This paper proposes an agent architecture, implemented by a newly devised SRLCS (Self-Reflection Learning Classifier System), that allows an agent to acquire a user model through interactions with the user and to adapt to changes in the user's internal states by using a sequence of interactions.

SRLCS provides *exploration* and *exploitation* as its learning mechanisms to infer the user's internal states. We conducted simulations on interactions between an agent with SRLCS and a postulated user model for two cases, where the user model has two and three different internal states. Through the simulations, it has been shown that SRLCS has great potential for the learning of interface agents.

Further studies, especially on how to effectively use *exploration* and *exploitation* and how to control the learning process, should be performed, as well as verifying the applicability of SRLCS through various agent applications.

References

[1] Cliff D, Ross S (1995), Adding Temporary Memory to ZCS. *Adaptive Behavior* 3(2):101-150

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

[3] Holland H (1985), Properties of the Bucket Brigade Algorithm. *The 1st International Conference on Genetic Algorithms (ICGA '85)* 1-7

[4] Sugimoto M (1999), User Modeling and Adaptive Interaction in Information Gathering Systems (in Japanese). *Journal of Japanese Society for Artificial Intelligence* 2(1):25-32

[5] Tosa N (1997), Art and Technology (in Japanese). *Journal of Japanese Society for Fuzzy Theory and Systems* 9(5):648-655

[6] Wilson S (1994), ZCS: A Zeroth Level Classifier System. *Evolutionaly Computation* 2(1):1-18

[7] Yoshida K (1999), User Modeling Methods for Intelligent Interface Systems (in Japanese). *Journal of Japanese Society for Artificial Intelligence* 14(1):11-16

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Interactive Self-Reflection Architecture Using Cellular Automata

○ Keiki Takadama
ATR Human Information
Processing Research Labs.
2-2 Hikaridai, Seika-cho, Soraku-gun
Kyoto 619-0288 Japan
keiki@hip.atr.co.jp
Tel: +81-774-95-1007
Fax: +81-774-95-1008

Koichiro Hajiri
Sony Computer Science Labs. Inc.
3-14-13, Higashi-Gotanda
Shinagawa-ku
Tokyo 141-0022 Japan
khajiri@csl.sony.co.jp
Tel: +81-3-5448-4380
Fax: +81-3-5447-1942

Katsunori Shimohara
ATR Human Information
Processing Research Labs.
2-2 Hikaridai, Seika-cho, Soraku-gun
Kyoto 619-0288 Japan
katsu@hip.atr.co.jp
Tel: +81-774-95-1070
Fax: +81-774-95-1008

Abstract

This paper proposes an *Interactive Self-Reflection architecture based on Cellular Automata (ISR-CA)* and investigates its adaptation ability through intensive simulations of multiple learning agents. Careful experiments of the adaptation ability has revealed the following implications: (1) ISR-CA provides a high level adaptability that cannot be obtained by using a conventional adaptation; (2) its adaptability is supported by an interaction between adaptation to the environment and adaptation from the environment.
Keywords: interactive, self-reflection, cellular automata, multiple learning agents

1 Introduction

When we consider the *adaptation* of agents, most people imagine a situation where agents adapt to given environments. Thus, a lot of adaptive systems are implemented based on such a consideration. However, Koestler claims that "in fact, the animal does not merely adapt to the environment, but constantly adapts the environment to itself [1]." He points out that (1) there are two aspects in adaptation, and (2) these two work as opposite aspects. Furthermore, the above claim may suggest that it is not sufficient for adaptive systems to employ only one aspect of adaptation.

However, some parts of the Koestler's claim are not clear when we implement adaptive systems according to his claim. First, there are several ways of assuming an environment. Second, there are also many ways of implementing adaptation mechanisms. To address both problems, we first focus on a multiagent environment that includes a lot of agents. This is because this environment embeds both aspects of adaptation, that is, the agent can adapt to the environment (including other agents) and can also adapt the environment to itself. Second, we employ the concept of *Interactive Self-Reflection (ISR)* [3] because ISR has both aspects of adaptation. Employing both the multiagent environment and the concept of ISR, this paper analyzes the characteristics of adaptation.

This paper is organized as follows. Section 2 starts by describing the concept of ISR and its model, and Section 3 presents our simulations. Finally, our conclusions are given in Section 4.

2 ISR and its Model

2.1 Concept of ISR

Interactive Self-Reflection (ISR) is a concept that tries to address more essential and fundamental elements for implementing the autonomy of agents in the true sense. In particular, ISR focuses on (1) boundary setting between self and others, (2) sense of values (SOVs), and (3) self-generated goal and evaluation criteria from the viewpoint of the "interaction" and "self-reflection" of agents (The details were reported in [3]).

2.2 ISR Based on Cellular Automata

Although ISR has a great potential for implementing both aspects of adaptation, ISR can be modeled in many ways due to its general concept. Therefore, we propose an ISR based on Cellular Automata (ISR-CA) as one concrete model. We select CA [4] because CA is suited for the concept of ISR in terms of the following reasons: (1) unpredictable and complex patterns created in CA are challenging date to be distinguished by *boundary setting between self and others*; (2) if *SOVs* are assumed as the way of thinking, SOVs can be roughly implemented by state transition functions*1 in CA; and (3) the property in which complex

* Paper submitted to the Fifth International Symposium on Artificial Life and Robotics (AROB'2000)

*1 Although we can consider a lot of functions in CA, one function is denoted as $a_i^{(t+1)} \leftarrow F(a_{i-1}^{(t)}, a_i^{(t)}, a_{i+1}^{(t)})$, where $a_i^{(t)}$

patterns are driven from simple state transition functions is useful for *generating both goal and evaluation criteria.*

2.2.1 Architecture

ISR-CA is composed of many agents, and each agent in this model has the same architecture, which includes the following problem solver and mechanisms as shown in Fig. 1. In this model, an environment includes an ideal self, other agents, an organization that agents belong to, and public opinions as outsiders*[2]. From this definition of environments, the agents in ISR-CA are assumed as the subject at certain times and the environment at other times. To distinguish this difference, Fig. 1 shows the boundary line between self and others. Furthermore, the aim of the agent is to acquire functions for an ideal self, other agents, an organization, and public opinions. This aim indicates the understanding of the four kinds of environments.

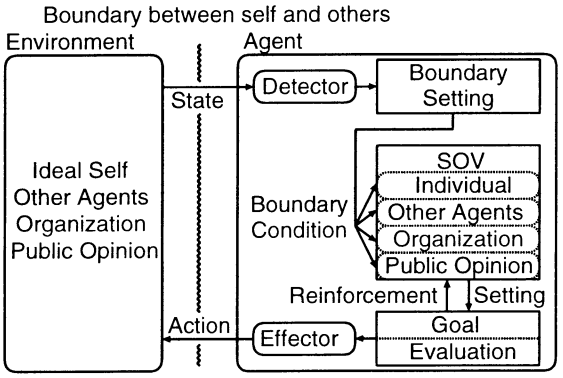


Figure 1: **ISR-CA Architecture**

< **Problem Solver** >

- **Detector and Effector** translate a part of an environmental state into the internal state of an agent and derive actions based on the internal state [2], respectively.

< **Mechanisms** >

- **(1) Boundary setting, (2) sense of values, and (3) goal and evaluation criteria** are key elements of ISR-CA (Details are described later).

indicates the state value of site i at time step t. Using this F, for example, No. 90 (= 01011010) function changes the state values as follows: 111→0, 110→1, 101→0, 100→1, 011→1, 010→0, 001→1, 000→0. Although the word *rule* is used instead of *function* in [4], this paper employs the latter word to distinguish the rule and function levels, and defines the function as a set of rules such as 110→1. This is because the way of thinking is not sufficient for one to one mapping at the rule level but must be implemented by a more global mapping at the function level.

*[2] This classification is mostly taken from research in [5].

2.2.2 Mechanisms

(1) Boundary Setting: In ISR-CA, the *boundary setting* mechanism precisely recognizes the current situation by determining the boundary between self and others. In detail, this mechanism recognizes whether the agents behave as an individual, other agents, an organization, or public opinions, and sets the boundary between self and others according to the recognized situation. For example, when the current situation promotes to behave as an organization, the agent sets the boundary that includes not only the agent itself but also other agents that belong to his/her organization. Furthermore, the situation that the detector recognizes as one of the environmental states is implemented by the state of CA whose number is `SITUATION_LENGTH`, and various situation are created by the pre-assigned function in CA.

(2) Sense of Values (SOVs): The *SOVs* mechanism in ISR-CA appropriately detects or produces the required information according to the SOVs. As shown in Fig. 1, the SOVs mechanism is generally composed of four functions for an individual, other agents, an organization, and public opinions. Although the maximum number of functions in SOVs is defined as 4, the initial number of functions is determined at random and the type of the function (*e.g.*, function for an individual) is also determined at random. With this setting, a certain agent may only have one function and other agents may have more than two functions whose types are the same except for the function number.

Basically, each function in the SOVs has a weight indicating its worth and this is initially set as `INITIAL_WEIGHT`. One function is probabilistically selected according to the weight of each function. In this selection, the weight of one function which is related to the recognized situation in the boundary setting mechanism is temporally biased by multiplying some values (`BIAS`), and this weight is reset as an original value after the function selection. For example, when the boundary setting mechanism recognizes that the agent has to behave as an individual in the current situation, the weight of the function for individual is biased in the function selection. This indicates a direct effect on to the function selection through the boundary condition as shown in Fig 1.

Furthermore, if there is no function which is related to the recognized situation in the boundary setting mechanism, the function is added to the SOVs up to a total of 4 functions. Especially when the agent has the maximum number (4) of functions, the function with the minimum weight is removed and a new func-

tion is added. After the above selection and creation of functions, each agent randomly creates an initial state whose number is INFORMATION_LENGTH, and generates a new state from this initial state according to the selected function. This self-generated state can be considered as information that is required for the agent.

(3) Goal and Evaluation Criteria: Finally, the *Goal and Evaluation Criteria* mechanisms in ISR-CA autonomously generate both the goal and evaluation criteria. In detail, the goal in this model is implemented by the state of CA whose number is GOAL_LENGTH, and a new goal is generated according to the function for individual when an agent achieves the current goal. Evaluation criteria, on the other hand, are determined not according to the generated goal but according to the recognized situation. This enables the agents to find various combinations of the goal and evaluation criteria.

As a concrete mechanism for the goal and evaluation criteria, each agent employs the same initial state created in the SOVs mechanism. The agent then generates a new state from this initial state according to the function which is related to the recognized situation, and compares this state with the state generated in the SOVs mechanism. For example, when the current situation promotes to behave as an individual but the agent selects the function for other agents, two new states are generated according to both cases and these states are compared. If both generated states are the same, a new goal is generated and the selected function is positively reinforced. If they are not the same, the current goal remains and the selected function is negatively reinforced. Concretely, the size of the reward is decided according to an equivalent degree of both states, and the decided reward is added or subtracted to the weight of the selected function in the success or failure cases[*3]. Especially in the failure case, the agent modifies the selected function by checking the different values in the state. This kind of learning is only done if the selected function is related to the goal[*4]. The agents' attitude as shown from the effector changes according to the reinforcement of weights and the goal setting as shown in Fig. 1,

2.2.3 Adaptation

As described in section 1, there are two aspects in adaptation, that is, (a) adaptation to the environment and (b) adaptation from the environment. In ISR-CA, one of these aspects is implemented by (a) goal generation and (b) self identity. Since it is not certain whether the generated goal will be appropriate or not in the current situation, the former mechanism enables agents to adapt to the environment by generating appropriate goals. Furthermore, since a frequent change in SOVs prevents an understanding of the characteristics of agents, the latter mechanism leads to adapt the environment (including other agents) to the agent itself by keeping its SOVs. In particular, self identity is assumed as to be the function for an individual in SOVs.

3 Simulation

3.1 Experimental Design

To investigate the characteristics of adaptation as described in sections 1 and 2, we tested the following three cases with three agents in our simulation.

- **Case 1:** Goal generation (Adaptation to environment)
- **Case 2:** Self identity (Adaptation from environment)
- **Case 3:** Integration of cases 1 and 2 (Both aspects of adaptation)

3.2 Experimental Setup

The following components in ISR-CA are designed for the concrete simulation.

- **Functions** are set as follows: functions for (a) an ideal self, (b) other agents, (c) an organization, and (d) public opinions are implemented by (a) the pre-assigned function, (b) the function in an individual of the neighbor's agent[*5], (c) the pre-assigned function, (d) the average of all agents' functions in an individual, respectively.
- **Reward** is calculated according to (the same number of values in a comparison of two states in CA) − (INFORMATION_LENGTH) + 1. This equation denotes that the reward becomes 1 in a perfect matching.

[*3] The weight must not be over the maximum weight (MAXIMUM_WEIGHT) and must not be below the minimum weight (MINIMUM_WEIGHT).

[*4] When the initial state is 100, the generated state according to the recognized situation is 010, and the generated state according to the selected function is 000, for example, the selected function is modified by changing the rule of 100 → 0 to 100 → 1 if the selected function is related to the goal. This indicates that the function number also changes according to this modification.

[*5] In detail, the function for other agents in agent(i) denotes the function in an individual of agent($i+1$).

- **Variables** are set as follows[*6]: `SITUATION_LENGTH` (the component numbers for for an environment) is 2, `INFORMATION_LENGTH` (the component numbers for an information generation) is 5, `GOAL_LENGTH` (the component numbers for a goal) is 2, `INITIAL_WEIGHT` (the initial weight) is 50, `MAX_WEIGHT` (the maximum weight) is 100, `MIN_WEIGHT` (the minimum weight) is 0, and `BIAS` (the bias for weight of the selected function) is 10.

3.3 Experimental Results

Fig. 2 shows the change in the averaged weight in the three cases[*7]. The values in this figure are calculated by using $(\sum_{n=1}^{3} ave_weight[n])/3$, where $ave_weight[n]$ denotes the averaged weight of all functions in the n agent. Furthermore, the vertical and horizontal axes indicate "averaged weight" and "iteration counts", respectively. In particular, one iteration is counted when all agents perform three mechanisms of ISR-CA. From this result, we find that the averaged weight in case 3 increases and becomes stable at high values, while the averaged weights in cases 1 and 2 decrease with oscillation.

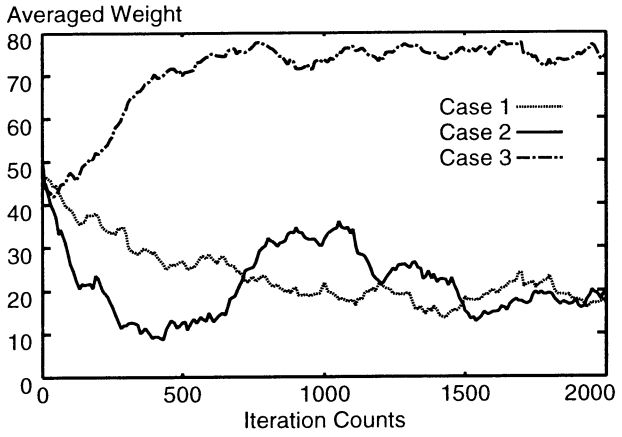


Figure 2: **Results of three cases**

3.4 Discussion

3.4.1 Adaptation to/from the environment

The averaged weights in cases 1 and 2 decrease with oscillation because of the following reasons: (1) the agents in case 1 cannot understand the characteristics (*e.g.*, the function number in ISR-CA) of other agents and public opinions due to the dynamical change of their characteristics; (2) the agents in case 2 cannot find an appropriate goal for the current situation because there is no goal generation mechanism. In case 1, agents can adapt to the environment by only using goal generation, if the environment does not change frequently. However, this assumption is not realistic in a multiagent environment. In case 2, on the other hand, the agent can adapt the environment to itself by only using self identity, if the evaluation criterion are determined according to the goal. However, this assumption is also not realistic because the evaluation criteria usually change according to the current situation.

3.4.2 Both aspects of adaptation

In a comparison of the results of cases 1 and 2, the averaged weight in case 3 increases and becomes stable at high values. This indicates that there is a limitation in the conventional adaptation like in case 1 or 2, and thus both aspects of adaptation must be considered in a realistic environment that includes (i) the dynamical change of an environment and (ii) frequently change of evaluation criteria. In particular, we claim that interaction between both aspects of adaptation is required in order to implement adaptation in the true sense.

4 Conclusion

This paper has proposed the *Interactive Self-Reflection architecture based on Cellular Automata (ISR-CA)* and investigated its adaptation ability through intensive simulations of multiple learning agents. The main results are summarized as follows: (1) ISR-CA provides a high level adaptability that cannot be obtained by using a conventional adaptation; (2) its adaptability is supported by an interaction between adaptation to the environment and adaptation from the environment. Future research will include an investigation of a relationship between the boundary setting and both aspects of adaptation.

References

[1] A. Koestler: *The Act of Creation*, Penguin Books, 1964.

[2] S.J. Russell and P. Norving: *Artificial Intelligence: A Modern Approach*, Prentice-Hall International, 1995.

[3] K. Takadama and K. Shimohara: "Interactive Self-Reflection Model: Aim and Overview", *The Fifth International Symposium on Artificial Life and Robotics (AROB'2000)*, 2000, to appear.

[4] S. Wolfram: "Statistical Mechanics of Cellular Automata" *Reviews of Modern Physics*, Vol. 55, pp. 601–644, 1983.

[5] T. Yamaguchi: "Interactive Self-Reflection based Reinforcement Learning for Multiagent Coordination", *The Fifth International Symposium on Artificial Life and Robotics (AROB'2000)*, 2000, to appear.

[*6] Note that the tendency in results does not drastically change according to the parameter setting.

[*7] This is result was obtained with a certain random seed, but we found that the tendency of the result did not drastically change even with different random seeds.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A Study on Modeling and Simulation of Human-like Pose of Robot Arms

Kunihiko OURA
Waseda Univ.
3-4-1, Ohkubo
Shinjuku-ku, Tokyo
169-8555, JAPAN
kuni@mn.waseda.ac.jp

Chinatsu SATO
Tokyo Denki Univ.
Ishizaka, Hatoyama,
Hiki-gun, Saitama
350-0394, JAPAN
chinatsu@hanalab.k.dendai.ac.jp

Izumi HANAZAKI
Tokyo Denki Univ.
Ishizaka, Hatoyama
Hiki-gun, Saitama
350-0394, JAPAN
hana@k.dendai.ac.jp

Abstract

This paper proposes a method to construct a model of the robot arm that expresses human-like pose appropriately. First, a model of the arm can be obtained by solving the inverse kinematics problem, where we take two angles of human pose in consideration. Then we construct the simulator of the arm from the model deduced by our method, which will show effectiveness of the method.

1 Introduction

In the field of welfare service robots, it is very important to imitate human actions appropriately, because people are often afraid of robots' nursing actions. In order to achieve them, the model that expresses human actions, or their cost functions that evaluate the actions should be defined, but they do not exist until now.

Though there are many works that deal with the welfare service robots, most of them do not aim at the 'human-like pose'. Moreover, the inverse kinematics problem, which calculates each joint angle from given position of the end of arm, generally results in the solution to nonlinear equation, and existence and uniqueness of the solution are not guaranteed.

We try to solve these problems, by introducing two variables: one is the angle between the upper extremity and horizontal surface, and the other is the angle that rotates from the right side of the body. With these angles in consideration, we will be able to limit the solutions, and realize the 'human-like pose' of the robot arms by giving these variables appropriately.

Therefore, the purpose of this paper is, to construct a model of the robot arm which imitates human-like pose appropriately, by finding the relation between the above two variables and other joint angles or coordinate of the upper extremity position. Finally, we construct the simulator of the arm from the model deduced by our method, which will show effectiveness of the method.

2 Problem Statement

We take link coordinates of the arm to describe arm's pose. These coordinates are shown in Fig.1, where Σ_0 is base coordinate frame.

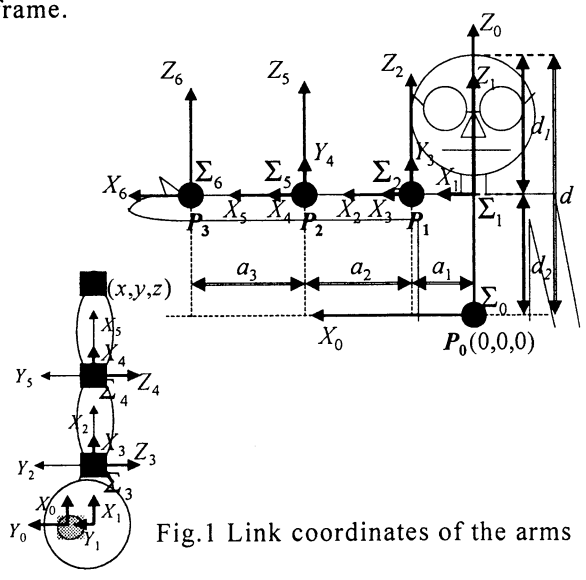


Fig.1 Link coordinates of the arms

It is assumed that a trunk, shoulder, upper arm and forearm are linked in the rotating joint, so that the arm has the five degrees of freedom. The arm's pose is completely described by joint angles θ_i (i=1,2,・・,5) .

The relation of wrist position in Σ_0 and each joint angle is defined by the following equations.

$$x = a_3(C_{12}C_{34}C_5 - S_{12}S_5) + a_2C_{12}C_3 + a_1C_1 \quad (1)$$

$$y = a_3(S_{12}C_{34}C_5 + C_{12}S_5) + a_2S_{12}C_3 + a_1S_1 \quad (2)$$

$$z = a_3S_{34}C_5 + a_2S_3 + d_2 \quad (3)$$

$$(C_i = \cos\theta_i, S_i = \sin\theta_i, C_{ij} = \cos(\theta_i + \theta_j), S_{ij} = \sin(\theta_i + \theta_j))$$

Here, a_1 is breadth of shoulder, a_2 is length of upper arm, a_3 is length of forearm and d is length of trunk. The problem to seek θ_i (i=1,2,・・,5) to realize the pose such that the wrist reaches to given position in coordinate Σ_0 is called inverse kinematics. But simultaneous equations (1),(2),(3) don't give unique equation.

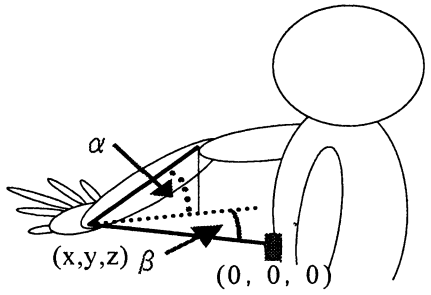


Fig.2 Angles taken from the human pose

We introduce two constraints to solve these equations. The one is an angle between the lower arm and ground, which is described by α, and the other is an angle between the wrist and navel , described by β. They are shown in Fig.2.

In this paper, we are trying to get the relation between wrist's position (x,y,z) and the angles α,β by measuring these quantities through natural upper limbs movement of the person.

3 The Solution of Inverse Kinematics with Constraints

When the angles α,β are given, we can solve the inverse kinematics and have the following results.

$$\theta_1 = \tan^{-1}\frac{h_y}{h_x} \pm \tan^{-1}\frac{o}{m} \quad (4)$$

$$\theta_2 = \mp\left(\pi - \tan^{-1}\frac{m}{o} - \tan^{-1}\frac{n}{o}\right) \quad (5)$$

$$\theta_3 = \tan^{-1}\frac{h_z - d_1}{\sqrt{a_2^2 - (h_z - d_1)^2}} \quad (6)$$

$$\theta_4 = \tan^{-1}\frac{h_z - z}{l} - \theta_3 \quad (7)$$

$$\theta_5 = \tan^{-1}\frac{l\tan\varepsilon}{\sqrt{l^2 + (h_z - z)^2}} \quad (8)$$

where (h_x,h_y,h_z) describe coordinates of the shoulder.

$$o = \sqrt{a_1^2 - m^2}\left(o = \sqrt{a_2^2 - n^2}\right), \quad \delta = \tan^{-1}\frac{x - h_x}{y - h_y}$$

$$m = \frac{a_1^2 - (a_2\cos\theta_3)^2 + h_x^2 + h_y^2}{2\sqrt{h_x^2 + h_y^2}}, \quad \varepsilon = \frac{\pi}{2} - (\delta + \gamma)$$

$$n = \frac{-a_1^2 + (a_2\cos\theta_3)^2 + h_x^2 + h_y^2}{2\sqrt{h_x^2 + h_y^2}}, \gamma = \theta_1 + \theta_2$$

$$\ell = a_3\cos\alpha\cos\varepsilon$$

From equations (4)-(8), we can limit the solution of inverse kinematics in two ways. Among them, we adopt the solution which imitates the human pose.

4 Simulation Experiment

In order to construct a model of human-like pose, we measure some joint angles of each point from the sitting person.

4.1 Experiment setup

We set two video cameras to take photos of human pose. One camera is installed over the examinees and the other is installed to the right side of them. The former is used for measurement of β, whereas the latter is for α. We put the paper with eight quadrangles of the size 10cm× 10cm on the table, which is used as

a guide for measuring.

In the experiment, we put marks on wrist, elbow and shoulder of the examinees, and measure a_1, a_2, a_3, d, d_1 and d_2 in advance. Each examinee puts his wrist inside every quadrangle one by one, and we take pictures of them with two cameras at the same time. We did it for three persons.

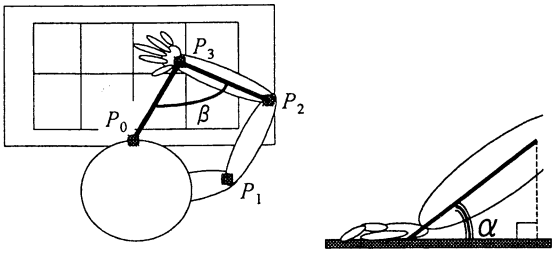


Fig.3 Angles taken from the experiment

After some photos are taken by two cameras, we measure each coordinate of $P_0 \sim P_3$ and the angle β, as shown in the left figure of figure3, and the angle α in the same way, as shown in the right figure of figure3. Here, coordinates are given with pixels and the angle β is measured by degree.

After the measurements, $P_1 \sim P_3$ is changed into a new coordinate system Σ_0 whose origin is P_0. We take two photos for each pose, for three persons. So that we have 24 data sets of (P_0, P_1, P_2, P_3) and (α, β).

4.2 Modeling of human-like pose

As we saw in the preceding section, the pose of the arm can be decided by angles α and β. So that we try to have a model of human-like pose by finding the relations between coordinates (x, y) and angles (α, β), when the examinees take natural pose.

At first, physical data are normalized as pretreatments. As the normalization, we assume $a_1 + a_2 + a_3 = 1$ about the arm length, and $d = 1$ about the height when sitting. We find a correlation coefficients between (α, β), (x, y) and each physical data.

Coefficients calculated from observed data are shown in Table1 and Table2.

Table 1 Correlation between α and other parameters

a_1	a_2	a_3	d_2	X	Y	α
1	-0.866	-0.981	-0.866	0.118	-0.317	0.018
-0.866	1	0.756	0500	-0.091	0.235	-0.267
-0.981	0.756	1	0.945	-0.120	0.237	0.077
-0.866	0.500	0.944	1	-0.113	0.315	0.236
0.118	-0.091	-0.120	-0.113	1	-0.245	-0.359
-0.317	0.235	0.327	0.315	-0.245	1	0.285
0.018	-0.266	0.077	0.236	-0.358	0.285	1

Table 2 Correlation between β and other parameters

β	a_1	a_2	a_3	d_2	X	Y
1	0.372	-0.175	-0.420	-0.467	0.830	-0.576
0.372	1	-0.866	-0.982	-0.866	0.118	-0.317
-0.175	-0.866	1	0.756	0.500	-0.091	0.235
-0.420	-0.982	0.756	1	0.945	-0.120	0.327
-0.467	-0.866	0.500	0.945	1	-0.113	0.315
0.830	0.118	-0.091	-0.120	-0.113	1	-0.245
0.577	-0.317	0.234	0.327	0.315	-0.245	1

From above tables, we can understand that not all the parameters are needed to represent α and β. α has strong correlation with a_2, d_2 X, Y, and β does with a_1, a_2, d_2, X, Y. Then we have come to make a model of human-like pose only with these parameters.

We have a model for α and β by using these data. The results are as follows ;

$$\alpha = -189.21 a_2 + 70.55 d_2 - 0.0095 X + 0.014 Y + 55.53 \quad (9)$$

$$\beta = -685.47 a_1 - 857.20 a_3 - 232.22 d_2 + 0.11 X - 0.11 Y + 700.12 \quad (10)$$

4.3 Construction of a simulator from the model

We evaluate the model by construction of a simulator for the arm. We use MATLAB to make them visualized. Figure4 shows the feature of a simulator.

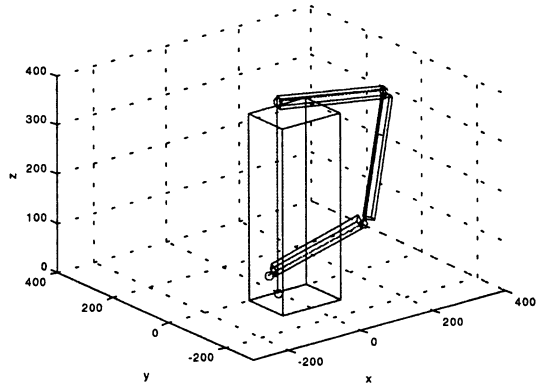


Fig.4 Simulator

If we give both initial and terminal points, the arm moves between them by solving inverse kinematics problem with α and β.

From simulation results, you may feel artificial movement on the shoulder, but on the behaviors of other parts, you may feel natural. The reason is that our simulator only assumes one degree of freedom in the shoulder, whereas in real case, it is said that it has two degrees of freedom.

5 Conclusion

In this paper, we built the model of the arm which imitated human-like pose appropriately. We took two angles of human body in consideration for the model, and constructed a simulator of it. Simulation results showed effectiveness of our technique.

References

[1] Kawasaki H , ロボット工学の基礎, Morikita (In Japanese)

[2] I.A.Kapandji, カパンディ関節の生理学、Ishi-yaku press (in Japanese)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Simplification of Visual Sensing in Skill-Based Manipulation

Akira NAKAMURA*
Takashi SUEHIRO*
*Electrotechnical Laboratory
1-1-4 Umezono, Tsukuba, Ibaraki, 305-8568 Japan
E-mail: nakamura@etl.go.jp

Tsukasa OGASAWARA**
Hideo TSUKUNE***
**Nara Institute of Science and Technology
8916-5 Takayamacho, Ikoma, Nara, 630-0101 Japan
***Oita Industrial Research Institute
1-4361-10 Takae-nishi, Oita-city, Oita, 870-1117 Japan

Abstract

Typical manipulator assembly tasks can be generally broken down into several motion primitives. We call these "skills" and explain how most manipulator tasks can be composed of skill sequences. Skills are also used to compensate for errors both in the environment model and in manipulator motions. By considering dispensable data, we can simplify the job of visual sensing. Furthermore, a simplified geometric model for visual sensing can often be derived. This paper describes an approach to simplifying visual sensing in skill-based manipulation.

Key words: manipulation skill, geometric model, visual sensing

1 Introduction

To be useful in several fields, it is necessary for a robot to achieve various tasks by using special techniques of manipulation. By analyzing human motions in tasks such as assembly and disassembly, we found that movements consisted of several significant motion primitives. We call these "skills" and have shown that most tasks of a manipulator can be composed of sequences of skills [1]-[4]. We demonstrated that robots are capable of performing various human tasks by using the concept of skill. This skill level control is positioned between task level control and servo level control in the manipulator control hierarchy. A programmer can describe a task program easily as a sequence of these skills, without taking servo level control into account.

Skills in which the contact states vary during assembly and disassembly tasks are particularly significant. We considered three skills, "move-to-touch," "rotate-to-level" and "rotate-to-insert," which play an important part in such tasks. For simplicity, we will explain skill motions in a two-dimensional environment. First, the move-to-touch skill is the transition from a "free" to "vertex-to-face" contact between the grasped object and another object. A similar transition, maintaining the contact in a different direction of motion, is also included in this skill. Secondly, the rotate-to-level skill is the transition from the "vertex-to-face" contact to an "edge-to-face" contact. Thirdly, the rotate-to-insert skill is the motion of rotating the object obliquely into a hole to complete the insertion.

Skill technique is also used to compensate for errors both in the environment model and in manipulator motion. The move-to-touch skill can cancel model errors in the direction of the transition, so it is not necessary to measure accurately in the direction of the motion. Similarly, the rotate-to-level skill and the rotate-to-insert skill can cancel model errors in the direction of rotation, so accurate measurement in the direction of rotation is not required either. Furthermore, we can derive a simplified geometric model for visual sensing by considering a maximum error or margin in a task that is composed of skill motions. For example, let us consider the task of unfastening a bolt using a wrench, which is composed of move-to-touch skills, rotate-to-insert skill and tilt motion. After attainment of touch skills, this task is always achieved by the rotate-to-insert skill, regardless of the bolt angle. In this task, it is not necessary to measure the orientation of the bolt. Therefore, hexa-head bolt can be regarded as a round shape in the step of visual sensing. Since the task is accomplished by measuring only the position of the bolt, visual sensing becomes easier.

In this paper, we consider a method for simplifying visual sensing in skill-based manipulation. We also propose a simplified geometric model for visual sensing.

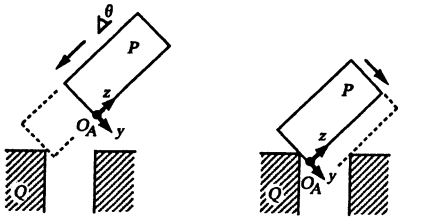
The next section explains manipulation skills. The dispensable and indispensable data for these three skills are detailed in section 3. The dispensable and indispensable data for tasks composed of multiple skills are explained using examples in section 4.

2 Manipulation Skills

This section explains our concept of skills. See References [1]-[4] for more detail.

2.1 Skill Primitives

In assembly and disassembly tasks, skills in which the contact states vary are particularly significant. We



(a) Move-to-touch$_f$ skill (b) Move-to-touch$_c$ skill

Fig. 1 Move-to-touch skills

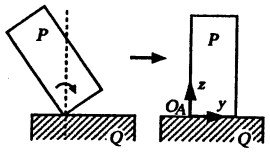


Fig. 2 Rotate-to-level skill

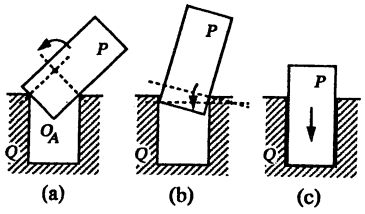


Fig. 3 Rotate-to-insert skill

will consider three skills that play an important part in such tasks: move-to-touch, rotate-to-level and rotate-to-insert. Most assembly manipulation tasks can be composed of these three skills. For simplicity, we view skill motions as occurring in two-dimensional environments.

(1) Move-to-touch Skill

The move-to-touch skill is the transition from a free to vertex-to-face contact between a grasped object P and another object Q in velocity control mode (Fig. 1(a)). A similar transition, maintaining contact in a different direction of motion, is also part of this skill (Fig. 1(b)). These two transitions are represented respectively by the move-to-touch$_f$ skill and the move-to-touch$_c$ skill.

(2) Rotate-to-level Skill

The rotate-to-level skill is the transition from the vertex-to-face contact to an edge-to-face contact (Fig. 2). This skill is performed with pressing force.

(3) Rotate-to-insert Skill

In an insertion task it is generally difficult to achieve the state shown in Fig. 3(b) directly when the clearance is small. The state in Fig. 3(a) is achieved first using other skills, such as the skill sequence shown in Fig. 1(a) and (b). The state in Fig. 3(b) is then accomplished by gradually raising the object while maintaining contact, as in Fig. 3(a). The rotate-to-insert skill is this motion of rotating object P obliquely into the hole in another object Q to insert it accurately. In our study, we assume that the rotate-to-insert skill also includes the pushing motion required to achieve the goal of the insertion task (Fig. 3(c)).

2.2 Skill Commands

We have conceptually represented these skill primitives as operation commands.

(touch_v w a): move-to-touch$_f$ skill of transition of velocity a along the w-axis. There is a possibility that the move-to-touch$_c$ skill of transition by sliding occurs after the initial contact, depending on the angle of the touch and the coefficient of friction.

(touch_p w a v b): move-to-touch$_c$ skill of transition of velocity a along the w-axis, with pressing force b on the v-axis to maintain constant contact. There is a possibility that the actual direction of transition is inclined due to the constraints of the contact. In such a case, we regard the w-element of the actual direction of transition as velocity a.

(rotate w c v b): rotate-to-level skill of rotation of angular velocity c around the w-axis, with pressing force b on the v-axis.

(insert w c v b): rotate-to-insert skill of rotation of angular velocity c around the w-axis, and pushing motion with pressing force b on the v-axis.

In these commands, w and v correspond to x, y or z. As described in this paper, we assume that a, b and c correspond to either a "+" or "-", not a specific value. For example, the commands of motion in Fig. 1(a) and Fig. 1(b) are represented by *(touch_v z -)*, *(touch_c y + z -)*, respectively. For simplicity, we also use notation without parameters such as *(touch_v)*.

3 Dispensable and Indispensable Data for Skills

In this section, we will consider the environment data necessary to achieve manipulation skills. The check of the data is performed just before manipulation skills are executed.

Let us explain the dispensable data and indispensable data for each manipulation skill. For simplification, we assume that the viewpoint of the camera corresponds to a reference point of an object P.

(1) Move-to-touch Skill by *(touch_v)*

The following three cases are taken into account.

(Case 1) When P must stick to Q

This is the case when an object P has to stick to the goal on the upper side of an object Q (Fig. 4(a)). We assume that the X, Y-axes of coordinates are positioned

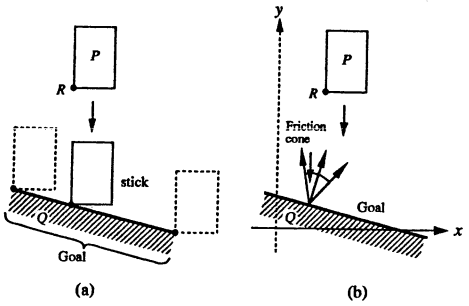


Fig. 4 Move-to-touch skill by (touch_v) (Case 1)

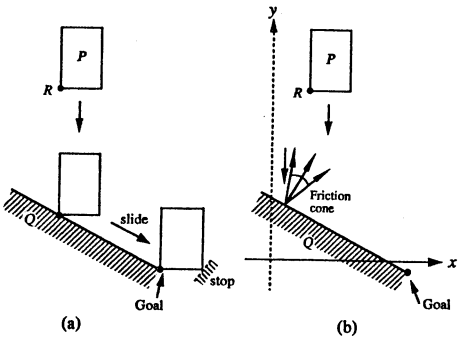


Fig. 5 Move-to-touch skill by (touch_v) (Case 2)

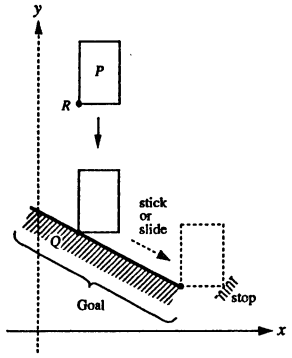


Fig. 6 Move-to-touch skill by (touch_v) (Case 3)

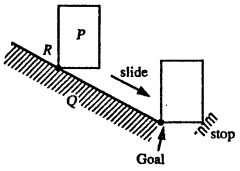


Fig. 7 Move-to-touch skill by (touch_p)

as the direction of motion of P corresponds to -Y (Fig. 4(b)). This skill is executed after the following conditions are checked (Fig. 4(b)).

(a) P is just over the goal of Q with respect to the X-direction.

(b) The direction of motion of P is included in the friction cone of the upper side of Q

The distance between P and Q does not need to be measured. That is, the depth value of the Y-direction is unnecessary.

Although the relative positions of P and Q may be ignored, the transition in the Y-direction is significant at the inside of each object P, Q. Therefore, we describe the Y-axis in Fig. 4(b) using a dotted line.

(Case 2) When P must slide on Q

Next, we consider the case in which an object P has to slide on the upper side of an object Q in order to attain the goal (Fig. 5(a)). We assume that the axes of

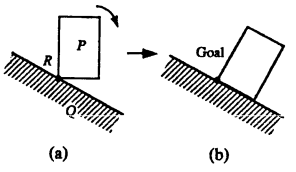


Fig. 8 Rotate-to-level skill

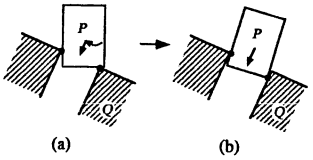


Fig. 9 Rotate-to-insert skill

coordinates are set similarly to (Case 1). This skill is executed after the following conditions are checked (Fig. 5(b)).

(c) P is just over the goal of Q with respect to the X-direction.

(d) The direction of motion of P is not included in the friction cone of the upper side of Q.

(Case 3) When P either sticks to Q or slides on Q

In this case an object P may stick to the goal on the upper side of an object Q as well as slide on the upper side of the object Q to attain the goal (Fig. 6). We assume that the axes of the coordinates are set similarly to the previous cases. This skill is executed after the following condition is checked.

(e) P is just over the goal of Q with respect to the X-direction.

In this case, the angle of the slant of the upper side of Q is not significant.

(2) Move-to-touch Skill by *(touch_p)*

This skill is executed after the following condition is checked (Fig. 7).

(f) P has a contact point on Q.

(3) Rotate-to-level Skill

This skill can be executed if an object P has the following relationship to an object Q (Fig. 8).

(g) P has a contact point on the upper side of Q as shown in Fig. 8(a).

This skill can compensate for model errors in the direction of rotation, so it is not necessary to measure accurately in this direction.

(4) Rotate-to-insert Skill

This skill can be executed if an object P has the following relationship to an object Q (Fig. 9).

(h) P has two contact points at the edges of the holes of Q as shown in Fig. 9(a).

In this skill, too, it is not necessary to measure accurately in this direction.

4 Dispensable and Indispensable Data for Tasks Composed of Skills

4.1 Tasks Composed of Multiple Skills

A manipulation task can generally be composed of

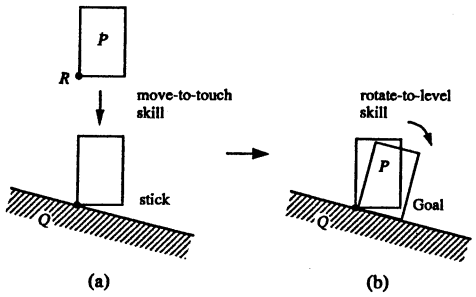


Fig. 10 Placing task

multiple skills. We will consider a sequence of skill primitives that compose the task. By applying the conditions in section 3 to the skill primitives of the task in sequence, initial conditions of the task can be derived.

4.2 Examples

(1) Placing task

We will consider the task of placing a box on a table, which is composed of the move-to-touch skill while sticking on a table and the rotate-to-level skill (Fig. 10). If a box P sticks to a table Q in the move-to-touch skill (Fig. 10(a)), the initial state of the rotate-to-level skill shown in (3) of section 3 is guaranteed (Fig. 10(b)). Therefore, this task is achieved if the relationship between P and Q is satisfied with conditions (a) and (b) in (Case 1) of section 3 (1).

(2) Task of unfastening a bolt using a wrench

Let us next consider the task of unfastening a bolt using a wrench as shown in Fig. 11, which is the same task in the example taken up in AROB 1st [5]. Assuming that the bolt is a movable object and the wrench is a fixed obstacle, opposite of their conditions in the real world, it becomes easier to consider conditions for achieving the task. We consider this task to be composed of the move-to-touch skill and the rotate-to-insert skill respectively using commands *(touch_v)* and *(insert)* (Fig. 11). The bolt-in-hole task can be always achieved for any angle of the bolt around the axis of rotation, using the rotate-to-insert skill. Therefore, this task is accomplished, even if the angle of the bolt is not known. The model of a bolt can be represented by a circle as shown in Fig. 12, in such a sense. Moreover, the initial condition of this task becomes similar to (e) in (Case 3) of section 3 (1).

5 Conclusion

We have described an approach to simplifying visual sensing in skill-based manipulation. We considered three skills, "move-to-touch," "rotate-to-level" and "rotate-to-insert," which play an important part in assembly and disassembly tasks. The dispensable and indispensable data for these three skills were explained in detail. Therefore, it becomes possible to simplify the job of visual sensing. We have also proposed a simplified geometric model for visual sensing. This is a robust model which is different from a real world object, taking skill-based manipulation into account.

In the future, we will further study simplification of visual sensing and simplified geometric models for various tasks. We must also take into consideration a realistic viewpoint which cannot be positioned on a grasped object.

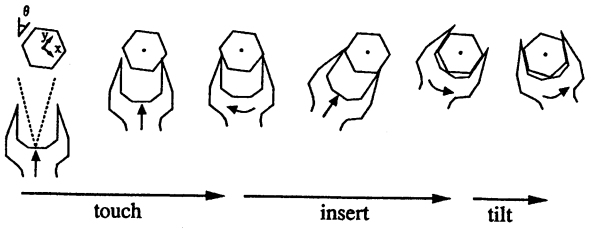


Fig. 11 Unfastening of a bolt by a wrench

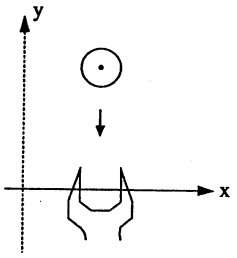


Fig. 12 Simplified geometric model

References

[1] Hasegawa T, Suehiro T, Ogasawara T, Matsui T, Kitagaki K, Takase K (1990), An integrated tele-robotics system with a geometric environment model and manipulation skills. In: Gotoh T, Kato I, Paul RP (eds) Proceedings of the IEEE International Workshop on Intelligent Robots and Systems (IROS '90), Tsuchiura, Japan, July 3-6, 1990, pp 335-341

[2] Suehiro T, Takase K (1990), Skill based manipulation system (in Japanese). J Robotics Soc Jpn 8:551-562

[3] Hasegawa T, Suehiro T, Takase K (1991), A model-based manipulation system with skill-based execution in unstructured environment. In: Dario P (ed) Proceedings of the 5th International Conference on Advanced Robotics, Pisa, Italy pp 970-975

[4] Hasegawa T, Suehiro T, Takase K (1992), A model-based manipulation system with skill-based execution. IEEE Trans Robotics Autom 8(5):535-544

[5] Nakamura A, Ogasawara T, Suehiro T, Tsukune H (1996), Fine motion strategy for skill-based manipulation. In: Sugisaka M (ed) Proceedings of the International Symposium on Artificial Life and Robotics (AROB 1), Beppu, Oita, Japan, Feb 18-20, 1996, pp 158-161

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

SlugBot: A Robotic Predator in the Natural World.

Ian Kelly, Owen Holland and Chris Melhuish.

Intelligent Autonomous Systems Engineering Laboratory, University of the West of England,
Faculty of Engineering, Bristol, BS16 1QY, England.
Ian2.Kelly@uwe.ac.uk

Abstract

A key aspect of the autonomy of living things is their ability to find and use sources of energy in the natural environment. Clearly, any comprehensive attempt at producing artificial life should demonstrate an equivalent capability; equally clearly, so should any truly autonomous robot. To date, both Alife agent simulations and robotic implementations have used environments and energy sources much too simple or structured to allow such equivalence to be claimed. This paper describes recent progress on an attempt to break free of these limitations by developing the world's first artificial predator – a robot which lives free on agricultural land, hunting and catching slugs, and fermenting the corpses to produce the biogas which is its sole source of energy.

Introduction

One of the most impressive facts about living creatures is that for much or all of their lives they are truly autonomous: they are able to survive and operate successfully in an unstructured environment without requiring any assistance whatsoever. Autonomy, whether biological or artificial, can be thought of as consisting of two major aspects: computational autonomy, and energetic autonomy. Computational autonomy refers to the ability to determine and carry out actions independently, whilst some of these actions may be related to the acquisition of energy, most will be concerned with other aspects of system operation. In the context of a biological system, the other aspects are associated with survival and reproduction, such as avoiding predators, finding shelter, grooming, finding mates, and so on. Energetic autonomy refers to the independent ability to maintain the internal availability of energy above the lethal minimum for sufficiently long periods to enable the system to achieve its mission, which in the biological context corresponds to securing the effective propagation of its genes. This ability does not merely involve making correct decisions in order to secure the raw energy; it also includes the conversion of the raw energy source into a usable form.

In the context of artificial life, a typical investigation of so-called 'autonomous agents' might involve simulated agents attempting to survive and reproduce in a world containing spatially localised elements corresponding to 'food' and 'predators', and so on. Although such abstract studies have been genuinely useful in exploring the dynamics of such situations, all fall very far short of the complexities faced by an actual animal in the real world. Perhaps the closest approach was made by Tyrrell [1], in a study within the area of adaptive behaviour rather than artificial life. Tyrrell reviewed and summarised the challenges faced by animals in surviving, and devised a simulation environment which included representatives of each class of problem. He then compared the various methods of action selection which had been proposed, in the contexts of biology and adaptive behaviour, as being able to support appropriate decision making within this environment. Three things are clear from his study:

- Existing artificial life simulations do not contain all of these problem types
- Even his environment is a gross simplification compared to any real environment
- None of the action selection systems examined comes close to the performance of any real creature

Over the last two decades, the design and control of autonomous robots has formed a major area of academic and industrial research. There are now many examples of robots or automated mobile systems (such as missiles, smart torpedoes, and some spacecraft) which achieve an apparently high degree of energetic and computational autonomy. Such systems carry enough fuel for their mission or can use radiant energy from their environment, and can control themselves 'intelligently' without human intervention. Some automated cleaning and materials handling AGVs use opportunity battery charging to achieve a degree of apparent autonomy. Several academic research groups have constructed robot environments which feature a 'powered floor', giving the possibility of indefinitely extended operation. However, on reflection, it is clear that most of these so-called 'autonomous' robots still require some explicit or implicit intervention from humans in order to carry out their tasks. Forms of human intervention include supplying information and energy, physically assisting the robot, and modifying the environment to suit the robot. The issue of autonomy has been finessed, rather than having been confronted and overcome.

This is by no means the first identification of the lack of autonomy in artificial agents – similar observations in a slightly different context were perceptively articulated by Steels [2] several years ago. However, the project described here represents the first serious attempt to design and construct a robot system with energetic and computational autonomy comparable to an animal system. We have sought to guarantee the comparability by constraining a robotic system to obtain its energy in

the same way as most animals - by finding and 'digesting' organic material in an unstructured environment.

This decision immediately brings a host of problems in its train. For instance, natural resources of this type are found only in certain types of places, and may only appear transiently, depending on time of day, season, and weather conditions; most importantly, the resources are destroyed by being used. The control of the foraging process must therefore take into account issues such as where and when to look for food, when to revisit an area where food was previously found, when to abandon a site which is not producing food, and so on. Once found, the organic resources, or food, need to be converted to an appropriate form of energy for storage and use. We propose to convert the organic material to electricity by using a mixture of biological and advanced engineering technologies: an initial and quite standard anaerobic fermentation process will be used to obtain biogas, and the biogas will then be passed through a specially developed tubular solid oxide fuel cell which generates electricity directly. Modern fuel cells of this type can produce electricity from biogas containing methane at concentrations of 20% or less [3].

The nature of the organic material to be used imposes further constraints. One option would be to use vegetable matter, which means that the robot would correspond to a grazing animal. While grazing is not a trivial problem, especially when resources are scarce, we did not think that such a choice would sufficiently challenge the requirement of computational autonomy. We therefore took the second option open to us: the source should be a mobile animal species, since the technical problems of predation are certainly severe enough from the computational viewpoint.

The choice of a prey species was guided by several considerations. Most obviously, the prey species should be reasonably plentiful and not require rapid pursuit, which would be difficult to achieve over soft ground at a reasonable energy cost. It is also likely that there would be some minimum energy cost associated with finding, catching, and consuming any individual creature, however large or small, and so the prey species should be large enough to give a significant margin over the minimum energy expended. Finally, in order to conform to ethical considerations, it should be an invertebrate pest species already subject to lethal control measures, so that the system would be doing something of actual use that would have to be done anyway. All of the above criteria are met by the slugs found on agricultural land in the UK, especially *Deroceras reticulatum* [4]. They are slow moving, abundant, relatively large, and extremely destructive - UK farmers spend over £20m per annum on buying and spreading molluscicides [5]. Slugs are also potentially more suitable for fermentation than some other possible target species: they do not have a hard external shell or exoskeleton and have a high moisture content.

Slugs are quite general pests, but perhaps do most damage to crops requiring a well cultivated seedbed, such as winter wheat or potatoes [6]. Following cultivation and planting, such ground is very soft; this means that a heavy fermentation vessel could not be moved over it without consuming large amounts of energy. The fermentation vessel/fuel cell system will not therefore be part of the robot, but will be in a fixed location, and the robot will bring slugs to it, and collect power from it. (This strategy is reminiscent of certain social insect colonies, where the insects bring raw organic material – which they cannot digest - to a 'fungus farm' in the nest, and feed on the fungus, which they can digest.) Inevitably, the fermenter will require a certain amount of energy to cover operating overheads, and our initial calculations indicated that a single robot would be unlikely to be able to gather sufficient energy to service both its own and the fermenter's energetic requirements; a multiple robot system must therefore be envisaged. Rather than being a handicap, this confers many potential advantages: for example, some search tasks might be performed more efficiently by several communicating robots than by a single robot. More importantly, the multiple approach gives a potential for achieving system reliability through redundancy. Interestingly, a robot that has run out of power away from the recharging station could be rescued by receiving power from another robot.

We have sketched out the basic concepts, and the logic behind the overall system design. The remainder of this paper describes the progress made to date in the design and construction of the robot hardware. Work on the fermenter and fuel cell system is to begin shortly, and initial field trials are scheduled for this year.

The Robots

To enable the robots to operate in a real unmodified environment and obtain their energy from the very limited resource represented by slugs, the robots will have to be energy efficient, reliable, and able to operate outside on soft and perhaps irregular ground. (They will also need to be protected from the weather, slug slime, and mud.) Energy efficiency can be achieved in several ways:

- by constructing the robots using light but strong materials like carbon fibre and aluminium
- by using decentralised modern low power controllers and electronics where possible - instead of a single high speed central processor - thus allowing currently unused devices to be shut down
- by the use of physical designs and control strategies designed to optimise efficiency

For example, the energy requirements for hunting and catching slugs have been minimised in the following way. Any system designed for finding and capturing slugs will require some means of searching the surface of the ground to detect slugs, and some means of picking slugs up from the ground. In our design, the sensor used

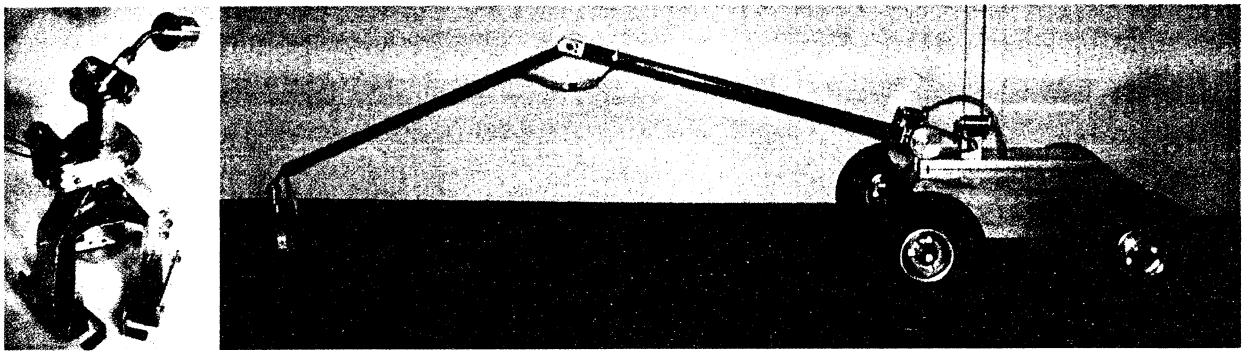


Fig. 1. Prototype three fingered gripper with wiper blades and compliance gimbal (left), and the arm and gripper system mounted on a turn table (right)

for detecting slugs, and the gripper used for catching them, are both located at the end of a long articulated arm. The energy required to move the end of this arm for a certain distance across the surface is much less than the energy required to move the whole robot through the same distance over a soft and muddy field. The robots will hunt for slugs by firstly moving to the centre of the region to be searched. The ground around the robot will then be scanned by rotating the arm around the body of the robot and gradually extending it, guiding the sensor, and hence the gripper, in a continuous spiral trajectory starting from the robot and moving gradually outwards. During this phase, the sensor and gripper will be actively maintained at a constant height above the ground. This spiral strategy covers the maximum amount of ground possible, and also reduces the power consumption of the scanning operation by conserving angular momentum. When a slug is found, it will be picked up by the gripper (which will already be in the correct location) and transferred to an on-board storage container. The arm will then be returned to its former location so that scanning can continue. When all the slugs at that location have been collected, the robot will move to a new location and repeat the procedure. (When the onboard slug container is full, or when the robot needs power, or when it appears that more energy will be used in hunting than is expected to be gained from it, the robot will return to the fermenter, where it will deposit any slugs and perhaps recharge itself.)

The optimal length of the arm is a function of the spatial density and distribution of slugs, and also of the power required to operate arms of different lengths and weights in relation to the power required to move the robot. Obviously, the longer the arm, the more ground it can cover without having to move the robot, but the more power will be required both for scanning and for picking up slugs. Our calculations showed that with between 1 and 10 detectable slugs per square metre, an arm with a nominal base length of around 1.5m would be the most efficient, but where there are many more slugs a shorter arm would be more efficient. On the basis of these results and some initial slug counts, we have opted for the 1.5m long arm. (At full extension, it measures 1.86m from the centre of the robot to the centre of the gripper.)

Other desirable attributes of the arm are that it should be light (and hence energy efficient), stiff, and easily controlled, and that it should be possible to manoeuvre the tip accurately. If possible, it should also have a simple construction, increasing its reliability, making it easier to manufacture and reducing its cost. (The total budget for the project, including labour and materials, is around $100,000.) The final version, designed and fabricated by Martin Scull of the Faculty of Engineering, consists of two 0.75m tubular sections, with a hinged joint between them (see Fig. 1, right). In order to allow the arm to rotate around the whole robot, it is mounted on a powered turntable located in the centre of the robot's chassis. The chassis is large enough to maintain stability in all directions when the arm is fully extended and loaded; the wheelbase dimensions were minimised by mounting the robot's batteries on the turntable on the opposite side from the arm, thus acting as a counterbalance. The arm is constructed from aircraft grade carbon fibre tube to meet the requirements of lightness and stiffness. To keep the arm structure light and the inertia low, the motor and gearbox required to provide movement at the elbow joint are mounted on the turntable; the drive to the elbow joint is transmitted by a lightweight toothed belt inside the arm. Since the numbers of slugs available on the surface peak in the early evening and just before dawn, the rate of gathering them during these periods must be as high as possible. To this end the arm motors and gearboxes were selected so that the arm can move from fully retracted to fully extended, or vice-versa, in under 1.5 seconds. Self locking worm gearboxes provide the required motor speed reduction, and allow the arm to be held in position without consuming any energy. All motors are fitted with optical encoders to provide position and velocity feedback.

The arm's end-effector is a robust lightweight gripper capable of picking up and releasing both wet and dry slugs, regardless of their size, sliminess, and orientation, and in the presence of irregularities in the substrate. Several design iterations were necessary; the final version consists of three fingers at 120 degrees spacing, operated by a single miniature motor (see Fig. 1, left). As the fingers close, they meet underneath the slug so that it can

be lifted; when the gripper is opened, wiper blades ensure the slug's release, however much slime is present.

Slugs are detected and targeted visually, and the camera carrying out this function is mounted in the centre of the gripper, away from slugs and mud, and ideally positioned to provide accurate feedback of the position of the gripper with respect to a target slug. To ensure that the view from the camera is always perpendicular to the ground, regardless of the arm's extension, the whole gripper assembly normally hangs freely on a gimbal. It is also possible to lock the gripper assembly in a fixed position, thus stopping any swinging, by fully opening the gripper. Underneath each of the three wiper blades is a plate which produces passive alignment with the contours of the ground when the gripper descends over a slug, ensuring that all three blades move under the slug when the gripper closes. The gripper mechanism will be protected from the weather, mud, and slime by a flexible cover.

Since the fermentation and recharging station will be in a fixed location, it is imperative that the robots are always able to locate and return to it. The nature of the terrain means that wheel slip and the irregularity of the substrate will preclude any extended use of odometry; we will take advantage of non-biological technology here by using a combination of the Differential Global Positioning Satellite (DGPS) system, and an active infrared localisation system [7], [8]. DGPS will also be used for mapping the locations of grazed areas, so that good patches can be found again, and over-grazing can be avoided. (In fact, this last point may not be a problem: a study undertaken for this project [9] found that the removal of all surface slugs from a field location every few days does not, in the medium term, appear to reduce the number of available surface slugs. It appears that a large reservoir of underground slugs is always present; as slugs are removed from the surface more slugs migrate upwards to replace them.

Although the fields in which trials will be undertaken are quite large and empty, obstacles may be present from time to time, and the robot will carry the normal complement of obstacle avoidance sensors required by any mobile robot. Obstacle detection will be achieved using a combination of ultrasonic sonar and, as a last resort, bump sensors. In addition, two sets of miniature ultrasonic sonar transceivers will be placed in the gripper: one set will point downwards so that the sensor for slug detection can be kept at the optimal distance from the ground regardless of any irregularities in the ground, and a second set will face outwards to detect any obstacles in the path of the gripper.

Detecting Slugs

We intend to operate the robots on crops of winter wheat, for about three months from the preparation of the seed bed. There are several reasons for this: it is the period when slugs are most numerous, when they are largest, and when they do most damage; it also offers us reasonably level ground with a relatively sparse crop cover. However, even under these relatively benign conditions, the detection of slugs is a difficult practical problem, mainly because of the presence of non-slug objects such as living and dying vegetation, stones, and lumps of soil. Many different types of sensors could be used for this task; we have opted for a vision based system since it offers the best combination of size, weight, cost, and effectiveness. VLSI Vision Ltd. produce a monochrome CMOS image sensor that is lightweight, relatively low power (<175mW), of adequate resolution (164 by 124 pixels), and sensitive (down to 0.1 Lux). It has a digital interface, and the maximum frame rate of 60 Hz enables reasonably high scan speeds for the arm. This image sensor also has adjustable automatic exposure control, and can calculate the average image intensity of the last frame, and perform pixel level thresholding using an adjustable threshold.

To reduce the complexity of the problem of the visual detection of slugs, we have developed a simple method of filtering out image components deriving from soil, living and dead vegetation, and stones. Since slugs are active mainly from dusk to dawn, some form of illumination of the imaged area will be required. This opens up the possibility of using a combination of coloured light and optical filtering to increase the relative visibility of slugs whilst decreasing the visibility of vegetation and earth. We have found that this can be achieved by using red light from extreme brightness LEDs, and placing a matching filter in front of the image sensor. Under these lighting conditions both vegetation and soil appear dark, whilst the slug *Deroceras reticulatum* reflects red light and thus appears very bright in the received images. Fig. 2 shows a 32mm long *Deroceras*, together with some grass, under white light - note the relative brightness of the slug and grass. Fig. 3 is the same image (except for some movement of the slug) taken using the red light and filter combination - the grass now appears dark whilst the slug is bright. Fig. 4 shows how the image of the slug can be picked out from the background by applying a simple pixel-based threshold function:

$$\text{Threshold} = (c[\text{Average Image Intensity}] - k)$$

where c and k are constants, to the red illuminated image of Fig. 3. There is an added benefit, in that the threshold scheme does not detect slugs under 15mm in length, and so filters out small slugs which in fact cost more in energy to retrieve than they can possibly yield. The final and relatively simple stage is the identification of bright patches which are of the correct size and shape.

As regards hardware, we use a low-power PAL to perform the high-speed image collection and compression; the data are then passed to a Scenix low-power 8-bit PIC-like microcontroller for shape recognition. The average image intensity and pixel-wise



Fig. 2. Under white light



Fig. 3. Under red light



Fig. 4. After thresholding

thresholding are both performed directly by the image sensor, with the Scenix microcontroller calculating the pixel threshold using the function given above. Each image is buffered into an 8-bit wide static RAM by the PAL, with eight thresholded pixels being stored in each location of the SRAM.

The Robot Control Architecture

The robot system is quite complex: there are several motors (gripper, elbow, shoulder, turntable, drive, steering) which must be precisely controlled, and several sensors (shaft encoders, imaging, obstacle avoidance, scan level, battery level, limit switches etc.) which must be monitored. Instead of centralising all processing, we have instead opted for a fairly distributed system with the emphasis on local processing, allowing us to use a low-speed three-wire serial (I^2C) bus to link all of the major subsystems. However, the high-level control of the robot is still handled by a single processor. By using a relatively decentralised approach, the overall system complexity is decreased, and the reliability should be increased. For example, all motion controllers share the same design – a PIC microcontroller handles communication and basic processing, and a Hewlett-Packard HCTL-1100 intelligent motion controller manages the shaft encoders and motor drives; each motor controller has a unique address on the I^2C bus.

Since the arm is capable of moving from fully retracted to fully extended (or vice-versa) in under 1.5 seconds, it is potentially hazardous, and we have had to confront some safety issues. Each closed loop motion controller incorporates a simple fail-safe protection system in the form of an independent watchdog. A separate retriggerable monostable controls the enable input of each motor driver. If a monostable does not receive a logic high to low transition every 20ms from its PIC microcontroller (due to a crash, or loss of communication) the associated motor will be shut down. In addition, both the PIC and the HCTL-1100 will stop the motor if it is driven into one of its limit switches. As final layers of defence, each motion controller board features motor current monitoring, and also has an emergency stop input capable of being activated by the high-level microprocessor as well as a stop command from a remote radio transmitter.

Control Strategy

Although the detailed engineering of the robot system presents enormous challenges, the deepest problems are expected to be at the level of high level control and decision making: when should the robot do what? This is the action selection problem examined by Tyrrell in [1]; it is well known within biology and adaptive behaviour, but has not been confronted by robot designers because the situation of a typical animal is radically different from that of a typical robot. In the industrial and military context, a robot or similar automated system usually has a single goal, and is rarely faced with a real choice of what to do next. In contrast, animals and our robots, which are in the same free-living open-ended situation, always have several simultaneous goals.

From the robot's point of view this includes such goals as: gathering slugs; recharging batteries; not becoming lost or stuck; always having enough charge to be able to return to the refuelling point; maintaining the functionality of its sensors and effectors by grooming.

Unfortunately, the action selection problem has not yet been solved: it is not known in general terms how to design a system which always selects the action which will maximise the expected survival time. Our only option, as designers and engineers, is to find and implement a computationally feasible solution which gives at least adequate performance. What makes the situation even more challenging is that, even with unlimited supplies of slugs, our system will at best be on the borderline of survivability, because our scheme for energy recovery is likely to be at least an order of magnitude worse than any biological system, and so the performance requirements are likely to be even more severe than those which an animal living entirely on slugs would face.

The chosen solution has been to adopt an existing model of motivation and action selection in animals and robots - the d,r,k model ([10], [11], [12]). This has been developed over several years, and is uniquely suitable for application on robots because it has been developed by considering animals as if they were robots, and vice-versa. It is computationally extremely simple – all that is necessary is the calculation at frequent intervals of a

numerical value for every possible action in the current situation. Although there are very many possible actions, the speed of a modern microprocessor makes it possible to perform such calculations very quickly. The factors involved in the calculation are also intrinsically simple: they involve the resources which the robot requires (slugs, cleanliness of sensors, knowledge of its location in relation to the fermenter, power in its batteries), the actions it can perform in the current situation to acquire or maintain each of these resources, and the rate at which the performance of each such action can be expected to produce the relevant resource. The main implementation problem centres on the last factor: an animal can generate accurate expectations through instinctive knowledge, or through learning. For example, an animal which lived on slugs would have very accurate expectations of how many slugs might be available in a certain type of location at a certain time of year under particular weather conditions. All we can give the robot at present is some very inaccurate expectations, and the hope that it will be able to learn to improve them.

Conclusions

Although significant engineering challenges remain, we have shown that it is possible to conceive of a credible plan for constructing a robotic predator, and that the execution of the initial parts of the plan has proved feasible. The project has already served to focus our thoughts on the real problems associated with true autonomy and self sufficiency in robots.

References

[1] Tyrrell T (1993), Computational Mechanisms for Action Selection. PhD Thesis, University of Edinburgh, UK

[2] Steels L (1995), When are robots intelligent autonomous agents? Journal of Robotics and Autonomous Systems 15, pp. 3-9

[3] Staniforth J, Kendall K (1998), Biogas powering a small tubular solid oxide fuel cell. Journal of Power Sources 71, pp 275-277

[4] South A (1992), Terrestrial Slugs: Biology, Ecology, and Control. Chapman and Hall

[5] Glen DM (1994), Ecology of Slugs in Cereals in Relation to Crop Damage and Control. In: Leather SR, Walters KFA, Mills NJ et al (eds), Populations, and Patterns in Ecology, Intercept, pp. 163-171

[6] Glen DM, Milsom NF, Wiltshire CW (1998), Effects of Seed-bed Conditions on Slug Numbers and Damage to Winter Wheat in a Clay Soil. Annals of Applied Biology, pp. 115, 177-190

[7] Kelly ID, Keating DA (1996), Flocking by the Fusion of Sonar and Active Infrared Sensors on Physical Autonomous Mobile Robots. Conf. on Mechatronics, pp. 1/1-1/4

[8] Kelly ID (1997), The Development of Shared Experience Learning in a Group of Mobile Robots. Ph.D. Thesis, Department of Cybernetics, University of Reading, UK, pp 2/4-2/12

[9] Glover C (1998), Reinvasion of Cleared Areas by the Slug Deroceras reticulatum. Internal Report. Long Ashton Research Station, Bristol

[10] McFarland DJ, Boesser T (1993), Intelligent Behaviour in Animals and Robots. MIT Press, Cambridge, MA

[11] McFarland DJ, Spier E (1997), Basic Cycles, Utility, and Opportunism in Self-Sufficient Mobile Robots. Robotics and Autonomous Systems 20, pp. 179-190

[12] Spier E, McFarland D (1998), Possibly Optimal Decision Making Under Self-Sufficiency and Autonomy. Journal of Theoretical Biology

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

The predictive operation of a micro robot

Masanori Sugisaka
○Yasuaki Ninomiya

Department of Electrical and Electronic Engineering, Oita University
700 Dannoharu, Oita 870-1192, JAPAN
msugi@cc.oita-u.ac.jp
ynino@cc.oita-u.ac.jp

Abstract

This paper surveys the recent trends in developing a micro robot soccer system and presents the design produce for soccer-playing robot. The robot soccer game has a lot of challenging problems and covers all the key areas of robot technology such as coordination between robot, motion planning of robots, visual recognition of objects, mechanics, sensors, machine learning and so on.

Our attempt is to develop a micro robot system with remote-brainless control architecture. The techniques include the chassis design, actuators, the radio link, the vision system software, and the control strategy software. Using a fast vision system, we obtain the configuration of each robot, and then the host computer computes the desired motion and commands each robot directly via RF-digital communication.

In this study, the attention is paid to the operation and the estimate of the robot orbit. The estimate of the control motion of the soccer robot and the orbit of object is described.

1 Introduction

It is the fact that it should take which act about the given situation is basic problem and especially, operation environment about the game makes the problem the more complicated one.

In robot soccer game, harmonizing, a decision, planning, learning and so on are set with image-algorithm, the recognition, the information communication of the robot and so on. Therefore, we study on the predication by using the soccer-robot.

2 The soccer robot system

The fact to choose hardware and software is the problem when controlling a soccer robot.

Mainly, as for the control method used, two of the soccer robot systems are

based on the sight and the robot. The former is used in this study.

Basically, the robots, the visual system, host-computer, the information communication system for the image processing are necessary for the soccer robot game.

2.1 The soccer robot system

There are two types of system based on the sight mainly.

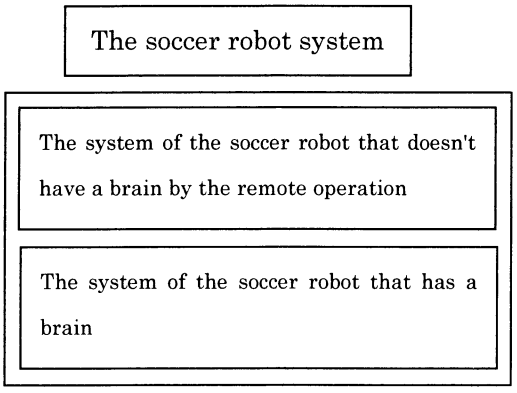


Fig.1 The soccer robot system

One is soccer robot system that doesn't have a brain. Another is the system that has a brain oppositely. The system that doesn't have a brain needs a host-computer when the robots work. On the contrary, the system that has a brain is the system that can judge whole situation by itself.

3 The operation of the soccer robot

3.1 The basic operation of the soccer robot

In the soccer game to cut a shoot and the ball from the enemy and so on, the basic movement of the soccer robot becomes more complex than that in the basic movement such as the running and the avoidance.

Also, the action in high level can choose higher-level control method for organization (the main control organization) at once.

3.2 The attack operation

The attack means leading to the shoots such as aiming at the goal or letting out a pass to the other robot.

Two relative positions are calculated when the information of the position of the ball and the goal can be given. Sometime one position is the position to begin shoot operation and another position is the position to complete shoot operation.

Shoot operation is executed when the following two conditions are satisfied.

① The ball is between the robot and the goal.

② The extension of the straight line which saw a ball from the robot is inside the range of the goal.

We suppose that these conditions describe positioning to take attack operation later.

3.3 The defense operation

Defense is operation to cut a ball contrastively in the attack operation or

companion robot to make save a ball. This operation is achieved by estimating an attack spot. We think of the position of the ball and the goal and moving this operation to the spot in the operation of defense-robot.

Because an attack route is decided by the position that the companion robot heads for, the position of defense-robot is influenced by the position of the ball and the position of the companion robot.

It is different from cut of the ball and this operation doesn't need the estimate of the orbit of the ball.

4 Simulation

4.1 The predication of the orbit of a ball

The fact to estimate the orbit of the ball connects with the movement to the optimal position by which robot itself can shoot the ball or can cut the ball. The orbit of the ball is estimated from both the present position and the described position of the ball. The speed is supposed in the constant in short time.

The ratio of distance that the robot runs to the position of the ball is the same as the speed ratio of the ball to the goal point can be computed by the following formula.

$$V_b \Delta t : V_r \Delta t = \sqrt{(Xi-Xb)^2+(Yi-Yb)^2} : \sqrt{(Xi-Xr)^2+(Yi-Yr)^2},$$

(Xb_0, Yb_0) shows the position of the present ball and (Xb, Yb) shows the predicated position of ball. $(X_r, Y_r), (X_i, Y_i), V_b, V_r, \Delta t$ are position of robot, the present position of the robot, the predicated position of the robot, the predicated speed of the ball, the speed of the robot and the time interval.

In the robot, because there is turning time when avoiding fault one and so on, a lag sometimes occurs to the goal point because of the information and so on.

As for the position computation, we can make a formula as much as the state which is secondary. The orbit of the ball is given by

$$AX^2 + 2BX + C = 0$$

where

$$R = Vb^2 / Vr^2$$
$$A = (R \cdot 1)(a^2 + 1)$$
$$B = R\{Xr + a(b \cdot Yr)\} + Xb \cdot a(b \cdot Yb)$$
$$C = R\{Xr^2 + (b \cdot Yr)^2\} \cdot Xb^2 \cdot a(b \cdot Yb)$$

It is possible to compute the direction where the ball moves from now.

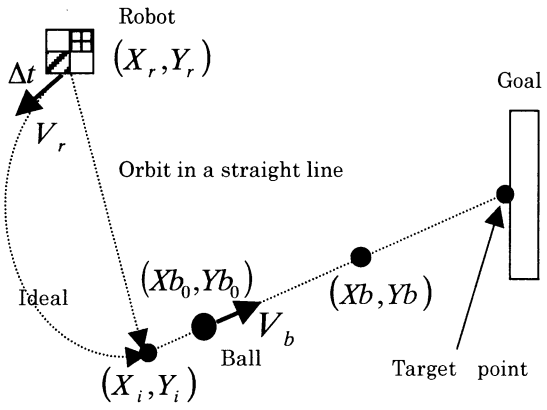


Fig.2 The orbit of the shoot

We suppose that the case to operate a program is considered according to Fig.2 shown above in order to agree with the ideal orbit approximately. We take

data in the limit that can be able to perform the case where "data A" doesn't meet with "data B". Also, we assume that the data A and B is not admitted as the data "data C".

A	B	C
47%	32%	21%

Fig.3 Simulation Result

5. Conclusion

It is expected that the robot has moved to the spot that is different from the shifted goal spot when mistaking the direction. Here the robot that was fixed at first by the visual system goes ahead.

However, in the experimental result, if supposing that the robot was working normally in the goal spot, it was possible to confirm that the difference between the ballpoint and the arrived point is small for some degree. It is difficult to estimate the exact ballpoint, which is moving when starting estimate. We have to consider a effective estimation in order for the robot to catch the ball when attempting to take an average.

The number of the operation in the unit time has to be increased. How to shorten the time to the goal point becomes a problem in the future.

6. References

[1] M.Sugisaka, 'The system theory and the control', in Japanese Nikkankou-gyou Sinnbunsya, 1997

[2] M.Sugisaka, 'Proceedings of The Third International Symposium on Artificial Life and Robotics', pp245-248, Beppu, Oita, 1998

[3] Sinzou Kitamura, Osamu Katai 'The picture and the control', Asakura Syoten, 1991

[4]M.K.Sahota,A.K.Mackworth,S.J. Kingdom and R.A.Barman, 'Real-time control of soccer playing robot using off-board vision: the dynamic test bed', IEEE Proc. Int. Com. Robots and Auto. , pp.68-82, Japan, 1995,

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Cooperative Action Controller of Multi-Agent System

Yong-Baek Kim
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
gione@hanmail.net

Hong-Min Jang
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
artrobot@hanmail.net

Dae-Jun Kim
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
djkimm@hanmail.net

Young-Kiu Choi
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
ykichoi@pusan.ac.kr

Sungshin Kim
Dept. of Electrical Eng.
Pusan National University
Pusan, 609-735, Korea
sskim0@pusan.ac.kr

Abstract

In this paper, we present a cooperative action controller of a multi-agent system and apply the controller to the soccer robots. To achieve a purpose, i.e., to win a game, it is necessary that a robot has its own role, action, and ability to work with one another. The presented action controller consists of three layers: the role selection layer, action selection layer, and execution layer. In the first layer, a fuzzy logic controller is used. Each robot selects its own action and makes its own path trajectory in the second layer. In the third layer, each robot performs its own action based on the velocity information which is sent from main computer. Finally, simulation and experimental results show that each robot selects proper roles and incorporates actions by the proposed cooperative action controller under a dynamic environment.

Keywords: Soccer robots, Multi-agent, Cooperative action controller, Fuzzy logic

1. Introduction

Multi-agent robotic systems are more flexible and fault tolerant since several simple robot agents are easier and cheaper to build compared to a single powerful robot for different tasks[1]. Micro-robot soccer is a suitable test bed for multi-agent system research[4]. The robots in a team have to cooperate, and at the same time they have to be in competition with the robots in the opponent team.

Recently, the robot soccer has been receiving considerable amount of attention. The robot soccer systems and control schemes have been also studied by many researchers with various degree of application and success[3,6].

Most of the previous control strategies developed for soccer robot system have been based on the modified zone defense strategy and special strategies for fouls[4]. However these control strategies bring some problems. That is, if a robot gets blocked by the opponent robot, the robot should stop, and if the ball is located in one of boundary areas, two robot can move towards it and may get collided with each other.

In this paper, we propose a cooperative action controller using fuzzy logic analyzer in stead of zone defense strategies. Due to analyzing the degree of opportunity for each robot and assigning a priority to each robot, this action controller overcomes some problems which take place when using zone defense strategies.

Simulation and experimental results show that each robot selects proper roles and incorporates actions by the proposed cooperative action controller under a dynamic environment.

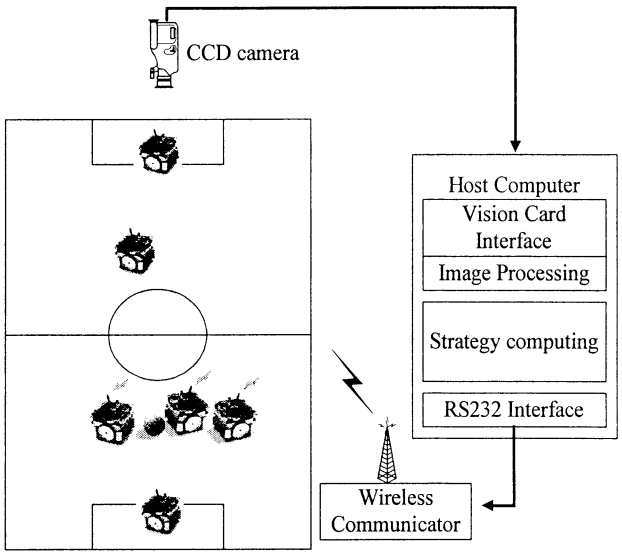


Fig. 1. The schematic diagram of robot soccer system

2. Robot Soccer System

The robot soccer system consists of three micro robots with the MiroSot(Micro Robot World Cup Soccer Tournament) specification[4], an external vision system with static configuration, and a host computer which supervises the robot soccer game as the central coordinator. Fig. 1 shows the schematic diagram of micro robot soccer system.

2.1 The Robots

Each robot is driven by two independent DC motors and sensorless velocity control. The soccer robot has 16 bit microprocessor Intel 80C196KC-20 on top to provide a fast PID servo loop for two DC motors. For wireless communication between the host computer and mobile robots, we use RF 418 and 433MHz communication modules. Fig. 2 shows the block diagram of mobile robot velocity control.

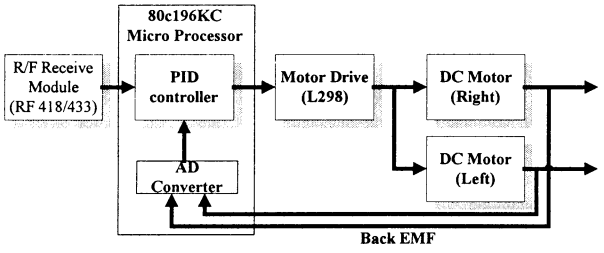


Fig. 2. The block diagram of the mobile robot velocity control

2.2 The Vision System

A color vision system is used to provide postion data of robots and a ball. The vision system consists of a CCD camera and a frame grabber interfaced with the host computer. The CCD camera provides NTSC signal output for the frame grabber. The frame grabber convert the raw data into the digital data at a maximum rate of 30 frames /sec. Each pixel on the 320 x 240 array of pixels in the image plane is assigned some YUV values of 8 bits data which are processed by image processing algorithm for object recognition.

2.3 The Communication Link

The RF module can be used in transmitting the continuous stream of information data from the centralized hierarchical controller to each robot playing soccer game. In the robot soccer systems, only the uni-directional communication link is implemented; the transmitter is located at the host computer, and the receiver is on the mobile robot.

3. Cooperative Action Controller

Our software for the robots has the hierarchical structure with three layers: the role selection layer, the action selection layer, and the execution layer. In the role selection layer, the role selection is performed by a fuzzy logic analyzer. The fuzzy logic analyzer analyzes the degree of opportunity of each robot using the current information and selects the role of each robot. Fuzzy logic analyzer collects training data through tentative game. Using these data, the fuzzy logic analyzer is trained by the steepest descent method. In the action selection layer, a proper action is selected based on the role chosen in the previous layer. According to the selected action, the path and trajectory planning are preformed, and the speed command for each robot is computed. In the execution layer, the speed command is transferred to each robot through wireless communication. PID control for DC motor is performed for each robot to follow the speed command.
Fig. 3 shows the block diagram of the cooperative action controller.

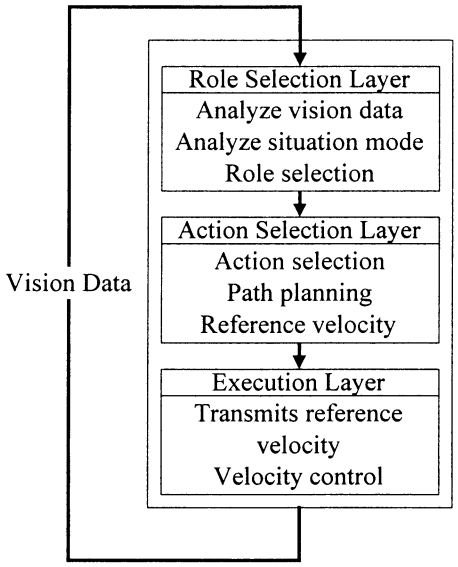


Fig. 3. The block diagram of the cooperative action controller.

3.1 The Role Selection Layer

From the vision data, we analyze robots' position, opponent robots' position and the ball's position. Next we analyze the situation mode. The situation mode is classified into two mode, i.e., an user defined mode and a playing mode.

■ Situation mode

1. User defined mode:
   At the beginning of the game, the robots do their role and action which is defined by user.
   Classification: Beginning as Offense, Beginning as Defense, Free Ball, Penalty Kick as Offense, Penalty Kick as Defense

2. Playing mode
Normal mode : using fuzzy analyzer they selects their role.
Emergency mode : some robots have fault and they overcome their fault.
Normal mode : each robot has its role from fuzzy analyzer.

The roles are defined as Goal Keeper, Attacker, Defender and Cooperator. They have preferential rights according to the ball's position. If the ball is in the defense region, the defender has more preferential rights than the attacker, and if it is in the offense region, the attacker has more preferential rights than the defender.

From the fuzzy analyzer, we estimate robot's opportunity, and the robot with higher opportunity than the other robots selects the role which has preferential right.

3.2 Fuzzy Analyzer

We can get each robot's opportunity from fuzzy analyzer except Goal Keeper. Fuzzy analyzer consists of the zero-order Sugeno fuzzy model (also known as the TSK fuzzy model) and has 2 inputs and 1 output. A typical fuzzy rule in the Sugeno fuzzy model has the form.

$$If\ x\ is\ A,\ and\ \ y\ is\ B,\ then\ \ z = f(x,y) \quad (1)$$

We use Gaussian membership function in its antecedent , and use constant in its consequent. Also we use the weighted sum operator and defuzzification with centroid of area method. Table 1 shows fuzzy rules used in the fuzzy analyzer.

Table 1. Fuzzy rules used in the fuzzy analyzer

		INPUT 1				
		NB	N	Z	P	PB
INPUT 2	NB	NB	NB	N	NS	Z
	N	NB	N	NS	Z	PS
	Z	N	NS	Z	PS	P
	P	NS	Z	PS	P	PB
	PB	Z	PS	P	PB	PB

NB: Negative Big , N: Negative , NS: Negative Small , Z: Zero, PS:Positive Small, P:Positive, PB: Positive Big

3.3 Action Selection Layer

In the action selection layer, each robot selects its action according to its role. Table 2 shows the actions based on the roles.
In the action selection layer, robot's action is chosen by a target circle and an action selection lookup table.

Table 2. Role and action

Role	Action
Goal Keeper	Home Goalie(HG), Inter Goalie(IG), Block Goalie(BG)
Offender	Attack(A), Shoot(S), Home Attack(HA)
Defender	Intercept(I), Inter Defense(ID), Home Defense(HD)
Cooperator	Sweep Ball(SB), Pass Ball(PB), Wait(W)

We define the target circle in Fig. 3. From the target circle, we can get a robot's relative position and assign a robot's action from the lookup table. Table 3 shows the robot's action from the relative position data.

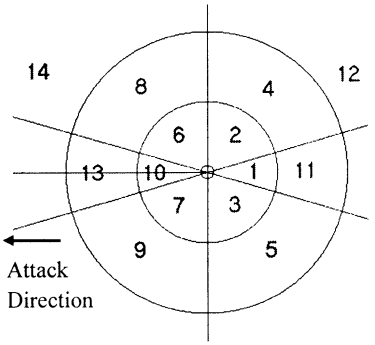


Fig. 4. Target circle

Table 3. Action selection lookup table

		ROBOT 1							
Position		2		4		6		8	
Opportunity		H	L	H	L	H	L	H	L
ROBOT 2	1	S	HD	S	HD	S	HD	S	HD
	3	A	HD	A	HD	A	HD	A	HD
	5	A	HD	A	HD	PB	W	A	HD
	7	A	HD	PB	W	A	HD	PB	W
	9	PB	W	A	HD	A	HD	A	HD

H: high, L: Low

The lookup table is defined by user and there are two types of tables according to ball's position. Table 3 shows the case that ball is in the offense region.

According to each robot's action, we can plan each robot's path. Given the initial position and direction, and final position and direction, we can construct a smooth path using straight lines and circles. After path planning, we can generate robot's reference velocity and send it to the execution layer.

3.4 Execution Layer

Host computer sends each robot's reference velocity using wireless communication. Then each robot receives its reference velocity and does velocity control.

4. Simulation and Experimental Results

In order to evaluate the controller, we have done simulations and experiments.

4.1 Simulation Results

We assume the following conditions.

Sampling time: 33 [ms]
Maximum velocity of robot: 1.2 [m /s]
Dimension of robot: 7.5x7.5x7.5 [cm^3]
Play ground size: 150x130 [cm^2]
Opponent robot uses local defense strategy as the action selection algorithm.

We play a game and record it. Then replay the game and gather the learning data which have wrong role selection. The fuzzy analyzer is trained by the steepest descent method. After learning we play games, and table 4 shows the results of the games.

Table 4. Game results

	Game No.	Score of Opponent Robots	Score of Our Robots
Before training the fuzzy analyzer	1	11	9
	2	11	10
	3	10	12
	4	12	8
	5	9	13
After learning the fuzzy analyzer	1	4	9
	2	4	12
	3	3	10
	4	3	12
	5	2	11

From the game results, we can find that the opponent robots and our robots have similar scores before the fuzzy analyzer is trained; however, after the fuzzy analyzer is trained, our robots are better than the opponent robots.

4.2 Experimental Results

We have done experiments on the environment which is the same as that in the simulation. Table 5 shows the game results.

Table 5. Game results

Game No.	Score of Opponent Robots	Score of Our Robots
1	3	10
2	1	9
3	4	13

From the game results, we can see that our controller is better than local defense controller. Our controller has merits that according to each robot's opportunity, each robot is assigned its role and select action. It is possible that each robot changes its role immediately and increases efficiency. That is, local defense algorithm assigns a region to only one robot, but our algorithm assigns all region to each robot. So there is a case such that our two robots compete with an opponent robot which covers some region. The duration that our robot gets the ball is longer than that of the opponent robot, so that our controller is better than the conventional ones.

5. Conclusions

In this paper, we proposed the cooperative action controller which consists of hierarchical structure with three layers. Role selection is performed by a fuzzy logic analyzer. The fuzzy logic analyzer is trained for optimal performances. We found that the proposed controller is better than the conventional controller through robot soccer experiments. Further study is necessary to include the information of the opponent team robots in the design of the more advanced robot controller.

References

[1] Y. Uny Cao, A.S. Fukunaga, and F. Meng, "Cooperative mobile robotics: antecedents and directions," *in Intelligent Robots and Systems*, pp. 226-234, IEEE Computer Society Press, August, 1995.

[2] Sukhan Lee and Javier Bautista, "Motion control for micro-robots playing soccer games," *in Proc. IEEE Conf. on R&A,* pp. 2599-2604, 1998.

[3] Arvin Agah, Brian Doyle, Matt Drees, Christian Froehlich, and Kelphen Kuok, "Robot soccer for the study of learning and coordination issues in multi-agent systems," *in Proc. IEEE Conf. on R&A,* pp. 3520-3525, 1998.

[4] 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," *in Proc. IEEE Int. Conf. on R&A*, Albuquerque, New Mexico, April, 1997.

[5] J.-S.R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-fuzzy and soft computing*, Prentice-Hall, 1997.

[6] Masanori Sugisaka and Xiaoshu Wang, "Evolution of multiple agent system from basic action to intelligent behavior," *KACC*, pp. 190-194, October, 1998.

[7] J.-H. Kim, K.-C. Kim, D.-H. Kim, Y.-J. Kim, and Prahlad. Vadakkepat, "Path planning and role selection mechanism for soccer robots," *in Proc. IEEE Int. Conf. on R&A*, Leuven, Belgium, May, 1998.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Labeling Q-Learning for Partially Observable Markov Decision Process Environments

Haeyeon Lee
Graduate School of Eng.,
Tohoku University,
Aoba05, Aramaki, Aoba,
Sendai, 980-8579, JAPAN

Hiroyuki Kamaya
Dept. Electric Eng.,
Hachinohe College of Tech.,
16-1 Uwanodai, Tamonoki,
Hachinohe, 039-1192, JAPAN
(yeon@abe.ecei.tohoku.ac.jp)

Kenichi Abe
Graduate School of Eng.,
Tohoku University,
Aoba05, Aramaki, Aoba,
Sendai, 980-8579, JAPAN

Abstract

The most widely used Reinforcement Learning(RL) algorithms, such as Q-learning and TD(λ), are limited to Markovian environments. Recent research on RL algorithm has concentrated on Partially Observable Markov Decision Processes(POMDPs). The only way to overcome partial observability is to use memory to estimate state. In [1], we proposed a new memory architecture of RL algorithms, called Labeling Q-learning(LQ-learning), to solve certain type of POMDPs and the LQ-leanrning was applied to test problems of simple mazes taken from literature. Simulation results demonstrated LQ-learning's ability to work well in near-optimal manner.

The LQ-learning is characterized by the extended information of observation. In LQ-learning, the agent attatches the labels to the observations; the agent receives more global information of environment than usual Q-learning. This method makes the agent learn its environment more effectively. The LQ-learning can solve the partial observation problems with more states than those tested in most previous POMDP works.

A key element in LQ-learning is labeling rule. In this paper, we propose three type of labeling rules only with small memory and give comparative study between these rules through experiments.

1 introduction

1.1 Reinforcement Learning(RL)

RL algorithm is one of the incremental learning algorithms. In the RL process, a learning agent receives a scalar reinforcement signal, called reward, as a feedback performance from its environment, and changes its parameters so as to act more optimally in the environment. In another words, the agent learnes the optimal policy that maximizes expected discounted rewards.

Environment is defined by a finite set of state S, and the agent has a finite set of actions A. In the RL process, the agent continually observes its environment's state $s \in S$, determines it's action $a \in A$ based on the observation and receives a scalar reward r. Then, the object of learning is to construct an optimal action policy that maximizes the expected discounted reward $E[\sum_{t=0}^{\infty} \gamma^t r_t]$ where r_t is the reward at time step t and $0 \leq \gamma \leq 1$ is discount factor that makes immediate reward more valiable than reward more distant in time.

In the widely used RL approaches, such as Q-learning, TD(λ), and Sarsa(λ) etc.(e.g., see[2]), the action selection at each step is based on Q-values which related to the goodness of actions. The Q-value is the total discounted reward that the agent receives if it starts from a state s, performes an action a, and behaves optimally thereafter.

In the RL approaches, the Q-values are based on the Temporal Difference(TD) method[3]. A standard form of the algorithms to estimate the Q-values with heuristic traces is expressed as follows,

$$\begin{aligned} Q(s,a) \leftarrow Q(s_t,a_t) \quad &+ \quad \alpha[\gamma_t + \max_a Q(s_{t+1},a_t) \\ &- \quad Q(s_t,a_t)]e(s_t,a_t) \end{aligned} \tag{1}$$

where α is learning rate, γ is discount factor, and $e(s_t,a_t)$ is eligibility trace. The eligibility traces emphasize more recent events more strongly. But we use a modification form of the eligibility traces, called the replacing traces[4].

$$e(s_t,a_t) \leftarrow 1 \tag{2}$$

$$e(s_t,a_t) \leftarrow \gamma\lambda e(s_t,a_t) \text{ for } s \neq s_t, \text{ or } a \neq a_t \tag{3}$$

1.2 POMDP

In many real problems, it will not be possible for the agent to have perfect or complete perception of the state of the environment. It can be considered that the agent makes observations of the state of the environment, but these observations may be noisy and provide incomplete information. This problem is also referred to as the problems of "incomplete perception", "perceptual aliasing", or "hidden state" and is formally modeled as POMDPs. In POMDPs, the mapping of the set of environment's states S to a finite set of obervations O is many-to-one. Thus many different states can be in the same perception. Many researchers have executed various problems of RL on POMDP. The only way to overcome partial observability is to use memory of previously occured actions and observations to disambiguate the current state.

A variety of approaches to PODMPs have been proposed[5][6]. However, such approaches are computationally complex and require large memory capacity.

2 Labeling Q-Learning(LQ-Learning)

2.1 Basic Concepts

The basic idea of LQ-learning is to introduce labels of the states. Basically LQ-learning is based on Q-learning of updating the Q-table and the action selection of ϵ-greedy policy. In the LQ-learning process, labeling information is attached to state observations, and a new perception of environment is generated. In detail, an observations $o \in O$ and a corresponding label θ in an element of finite set Θ, make a new perception of the environment, $\tilde{o} \in \tilde{O} = O \times \theta$. For these new perceptions, Q-value in LQ-learning is represented by $Q(\tilde{o}, a)$. Of course, the number of labels is restricted by scale of environment

2.2 Labeling Rules

At each step t, when the agent gets an observation $o_t \in O$ from the environment, a label $\theta_t = \theta(o_t)$ is attached to this observation, where θ_t is an element of a finite set $\Theta = \{0, 1, 2, ...M-1\}$. Thus, the pair $\tilde{o_t} = (o_t, \theta_t) \in \tilde{O} = O \times \Theta$ defines a new observation. The rule of labeling is defined by using a mapping $h : \tilde{O} \times \tilde{O} \rightarrow \Theta$. This mapping h works like a Hash function, and we call this function labeling function.

At the beginning of each experiment, the probability p is set to a positive number, $p_0(o_0)$, a little less than 1.0 for all $o_t \in O$. This probability increases gradually in LQ-process. $p_0(o_t)$ is the probability of changing label at observation o_t[1].

In this paper, only three examples of labeling rules are introduced.

$$\begin{aligned} \theta'(o_t) \leftarrow\ & h_1 \quad (\tilde{o}_t, \tilde{o}_{t-1}) && (4)\\ =\ & \max(\min(\theta(o_t)+1, M-1), \theta(o_{t-1})) \\ \leftarrow\ & h_2 \quad (\tilde{o}_t, \tilde{o}_{t-1}) \\ =\ & (\theta(o_t) - \theta(o_{t-1})) \bmod M && (5)\\ \leftarrow\ & h_3 \quad (\tilde{o}_t, \tilde{o}_{t-1}) \\ =\ & (\theta(o_t)^2 - \theta(o_{t-1})^2) \bmod M && (6) \end{aligned}$$

3 Task & LQ-learning Algorithm

3.1 Task Representation

The agent's task is to discover a path leading from start position to goal in a partially observable maze. The agent is assumed to have lifetime, T_{max}, separable into "trials". Each trial consists of at most S_{max} discrete time steps. Each trial ends either if it arrives at the goal within S_{max} steps,or if it can not find the goal in less than S_{max} steps. In each trial, once the agent arrives at the goal it receives a reward r=10.0. Otherwise the reward is zero.

3.2 Entire Algorithm

The entire LQ-learning algorithm is given below.

---

①Initialize the Q-table,latest success Q-table, $p(o)$.
②For $(T < T_{max})$
③ Initialize Label-table.
④ For step$(s < S_{max})$
⑤ Observe current environment.
⑥ (Attatch a label to o_t)
⑦ If $o_t \neq o_{t-1}$
⑧ $\theta(o_t) \leftarrow h(\tilde{o}_{t-1}, \tilde{o}_t)$ with probability $p_0(o_t)$
⑨ $p_0(o_t) \leftarrow 1 - \rho(1 - p_0(o_t))$ $(0 \leq \rho 1)$
⑩ Else $(o_t = o_{t-1})$
⑪ $p_0(o_t) \leftarrow p_0$
⑫ Select action.
⑬ Execute selected action.(ϵ-greedy policy)
⑭ Receive rewards
⑮ Update Q-table
⑯ If (s=Goal)
⑰ Go to ②
⑱ Else
⑲ Go to ④

This algorithm is for H2 and H3(see [1] for H1).

4 Simulations

4.1 Environment; Grid-World

In this paper, we accept Grid-Worlds as test problems of POMDPs. Grid-World is not restricted in arrangement of obstacles, starting points, and goal points, so huge complexity may be realized in the Grid-Worlds. Fig.1 is the example used in our simulations.

In the Fig.1, the black cells mean obstacles. Each

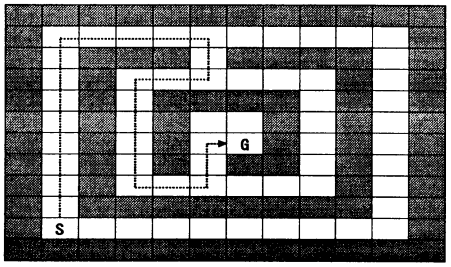


Figure 1: Grid-World used in simulations

state transition is a movement from one white cell to a next white cell. And in Fig.1,"S" means the starting point, and "G" the goal. The object of the agent is to reach the goal in as few steps as possible. The dotted line in each figure shows the shortest path, which the agent wants to learn. At each cell, the agent can only see which of the adjacent 4 cells are obstacles. The value of the north, east, south and west black cells next to the agent are 1, 2, 4, and 8, respectively, and the agent's observation is the sum of values of black cells surrounding the agent. Thus, there are 16 possible observations.

4.2 Action Selections; ϵ-greedy

In our simulations, only four actions are possible in a state-transition. Like the observations, the possible actions are only "go north", "go east", "go south", and "go west". But if the east cell is blocked, the "go east" is eliminated from the four possible actions.

We select actions by ϵ-greedy rule that involves the random action selections. The ϵ-greedy rule means that the agent selects the action of the maximum Q-value with probability 1-ϵ and instead selects an action at random, uniformly with a small positive probability ϵ. In addition, value of epsilon is decreased in proportion as trial proceess. At a trial T, the value of epsilon is given as follows :

$$\epsilon_T = (1 - \frac{T}{T_{max}})\epsilon_{max} \quad (7)$$

Here, ϵ_T is the epsilon at trial T, and the maximum value of epsilon is fixed at 0.1 in our simulations. So, for instance, if T_{max}=10,000, the value of epsilon is 0.01 in the 1000th trial.

4.3 Latest-Success Memory of Q-Table

In previous paper[1], we adopted the memories of success Q-table. In detail, at a trial, the agent arrives at goal, its Q-table is stored in the latest-success Q-table. And at the goal, the success Q-table is updated by its Q-table, and the next trial is started with the latest-success Q-table. The rule of memorizing the Q-tables is as follows.

At every trial, the agent has the following 3 Q-tables .

$Q_{current}$: Q-table at last step of current trial
Q_{latest} : Q-table at latest success trial
Q_{start} : Q-table of starting new trial

If a trial is finished in goal, the updating of Q-tables is the following :

$$Q_{latest} \leftarrow Q_{current}$$
$$Q_{start} \leftarrow Q_{current}$$

Otherwise, update Q-tables under the following rule :

$$Q_{start} \leftarrow Q_{latest}$$

(In this case, the $Q_{current}$ is destroyed)

---

5 Results

In our simulations, T_{max} is fixed at 100,000, S_{max} is at 100, and all results are presented by average of 40 simulations. We use the following parameters for all simulations : α=0.1 and γ=0.95. Grid-World has double walls inside the environment. The optimal step represented by dotted line is 27.

By each labeling function, the agent can learn the nearly optimal path. Of course, enormous kinds of labeling rules can be possible but we adopt here only three examples of labeling rules.

As is shown in Fig.2, Fig.3, the result of H1 looks

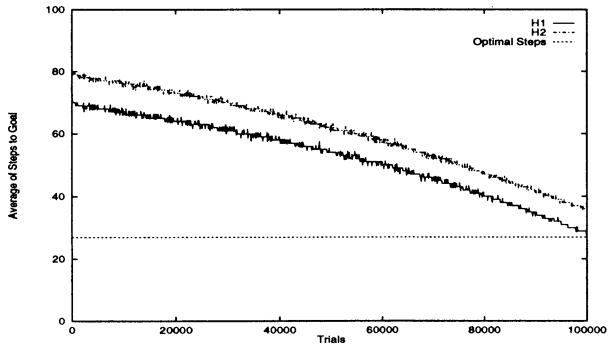


Figure 2: Result of H1 and H2

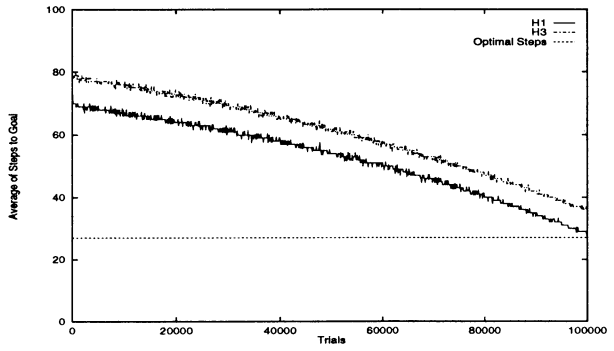


Figure 3: Result of H1 and H3

most effective, but H1 is constructed by experiences by trials and errors. To contrast with H1, we tested another two labeling rules, H2 and H3.

H2 is rather simple function. This is defined by the difference of observations of adjacent two states modulo the number of labels. From the result shown in Fig.2, the agent can learn the sub-optimal path, though not optimal.

Next, we showed the result of the function H3. In the view of the success of H2, we try the function H3, $({o_t}^2 - {o_{t-1}}^2)$ modulo M. However, we could not discover the difference between H2 and H3 in efficiency. It seems that the $(o_t + o_{t-1})$ term is unvaluable in this learning process.

6 Conclusion

From the results shown above, we could certificate the efficiency of the LQ-learning algorithm. Our experiments involve three type of labeling rules. All of these work well with small memory, though they are not optimal.

However, in LQ-learning we must consider about several factors to effect learning process. There is no doubt that the labeling rules have an important effect on Learning performance. And we used the memory of latest-success Q-tables. By limitation of computational time or capacity, we only store the information of the latest success Q-table. But, if more memory of past experiences is permitted, the efficiency of learning would be increased. Thus the memory structure is also one of the key factors to solve POMDPs. Finally, the number of labels is also important. It would be interesting to study adaquate number of labels, and how sensitive it is in learning process. Our future study is to construct an agent that automatically learn the optimal needed number of labels in LQ-learning process.

References

[1] Haeyeon Lee, Hiroyuki Kamaya, and Kenich Abe. Labeling Q-learning for Non-Markovian Environments, IEEE International conference on Sustems, Man, and Cybernetics, 487-491, 1999.

[2] Sutton, R. S. and Barto, A. G. Reinforcement Learning : An Introduction, Cambridge, MA, MIT Press, 1998.

[3] Sutton, R. S. Learning to predict by the methods of temporal differences, Machine Learning, 3, 9-44, 1998.

[4] Singh, S. P. and Sutton, R. S. Reinforcement Learning with Replacing Eligibility Traces. Machine Learning, 22, 123-158, 1996.

[5] Kaelblingm, L. P., Littman, M. L. and Moore, A. W. Reinforcement Learning : A Survey, Journal of Artificial Intelligence Research 4, 1996.

[6] Wiering. M. and Schmidhuber. J. HQ-Learning, Adaptive Behavior 6:2, 1997.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Multi agent simulation of Sika deers with GIS data

Go Yamamoto , Kazuko Yamasaki , Keitaro Hara , Yasuo Kinouchi
Department of Information Science
Tokyo University of Information Sciences
Tiba ,265-8501,Japan,go@rsch.tuis.ac.jp

Abstract

Using multi-agent simulator Swarm, We simulate population of Sika deers, which inhabit in the Kinkazan Island The landscape of Kinkazan and the distribution of Sika deers are given by GIS(Geographic Information Systems) in the every cell.

Every Sika deer decides his action by the decision making function, which involves current and neighborhood cell's landscape data.

And we derive suitable parameters of this decision making function using GA with the fitness which represent the proximity between the census of Sika deers in this island and computer simulation.

1 introduction

Kinkazan Island, which is a stage of the simulation, is the island which exists in the open sea in Miyagi Prefecture, and is respected as the sacred place. Therefore, houses, etc. do not exist except for a part of shrines, etc. And it is known as the island which is suitable for the census of Sika deer and the data such as the distribution of the Sika deer is taken.

One of the applied fields of multi-agent system is the field using GIS.

Existing works with GIS are the simulation concerning the ecology of fish school reading real landscape of river[1] and the simulation concerning the formation of a village in Northern America for 400 years from 10 centuries[2] and etc., which show the worthwhile, result in the ecology and sociology.

And , as the simulation of Sika deers , they are the simulation of Sika deers for the flood control in one region of Florida[3] and the prediction, not being multi-agent simulation, of the distribution of Sika deers extracting from ad@hoc rule base with cell data in the Grampian region[4].

But in all their works, the parameters of the decision making function of agents (fish, man and deer) are exogenous and given in the transcendental. On the contrary to them, we make attempt to introduce these parameters rationally. That is we derive suitable parameters of this decision making function using GA with the fitness which represent the proximity between the census of deers in this island and computer simulation.

The definition of distance between the census of deers and computer simulation is not self-evidently. To define distance between cell space distribution, in such case that each cell strongly depend on near cells, we cannot treat cells independent and use Euclid distance of the dimension of the number of cells. In this case ,the distance of the location in which deer exists near likely cell must be small. To describe suitable distance, we take time average of Euclid distance over long steps enough for deers to go around all near likely cells.

Our originalities of this work are following two points.

1. Use the adapted decision making function by means of GA.
2. Use the suitable distance between space distributions by means of time average of Euclid distance over long steps in which the distributions are thought to be identical.

2 model

In this simulation ,we describe landscape as mesh data(cell), The configuration of Sika deer and landscape are given in each cell. The size of cell is 30 square m , and Kinkazan island is covered by east and west 122 north and south 167 cells.

1. Each cell has following attributions.
   - 9 grade of elevation
   - 9 grade of angle
   - 9 grade of direction

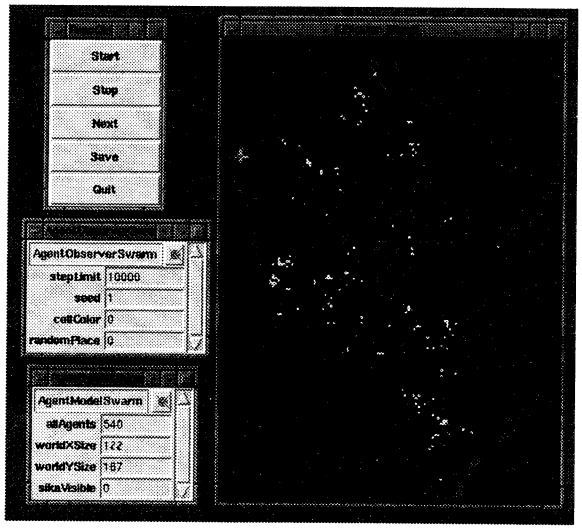


Figure 1: Sika deers on Kinkazan island

- 5 sort of land cover(bare, grass, composed of conifer trees, the mixed forest of conifers and deciduous broadleaved trees, the deciduous broadleaved forest which consists of Fagus crenata and Carpinus tschonosi etc.)

2. We distribute 270 Sika deers the same number as census data, in the cell allowing duplication.

3. A Sika deer can either move 1 cell on one direction or stay per a step. A Sika deer chooses his action among these 5 alternatives. A Sika deer's decision-making function is as follow.

   This describes cell's favorability. A Sika deer chooses the next cell stochastic in the proportion of cell's favorability.

4. All of Sika deers have the same decision making function in one simulation consist of 1000step.

   One simulation corresponds to one chromosome, each gene in the chromosome corresponds to one parameter of decision making function.

   Then one generation consists in simulations in the number of chromosomes .

   Simulation takes 100 generation ,each generation has 30 chromosomes , which are made 35 genes as real number being the parameters of decision making function of Sika deers.

5. Fitness function is calculated by cumulating the Euclid distance between the census of Sika deers and simulated temporary Sika deers' location as cell being independent over one simulation.

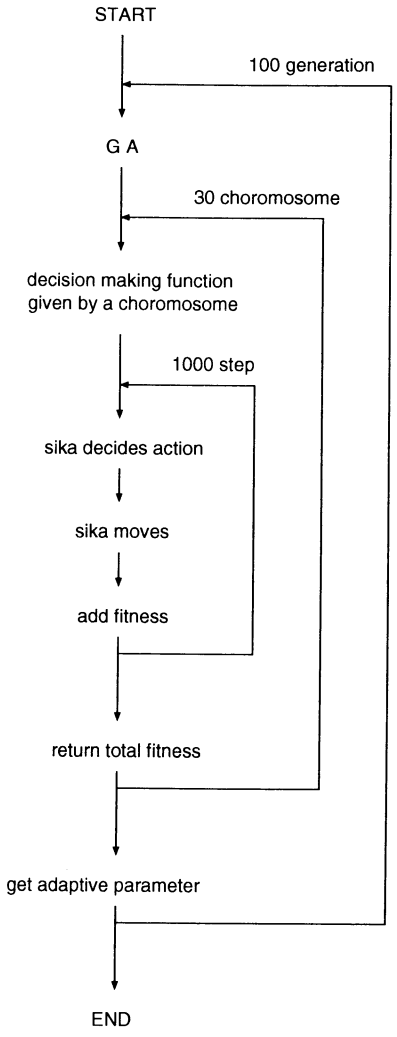


Figure 2: program flow

3 result

We initially distribute Sika deers as census data, and simulate these cases,

1. In the case of the adapted decision making function by GA.

2. In the case of averaging over the ten decision making functions with random parameters.

3. In the case that the ten decision making function's parameters are chosen in proportion to Sika's densities in each level of each cell's attribute.

We compare three results.

Figure 3 shows fitness in each step in three cases. The fitness remains high value over 3000 steps in the case of the adapted decision making function by GA. On the contrary in other cases ,The fitness declines rapidly. This means the distribution of Sikas is growing far from initial one.

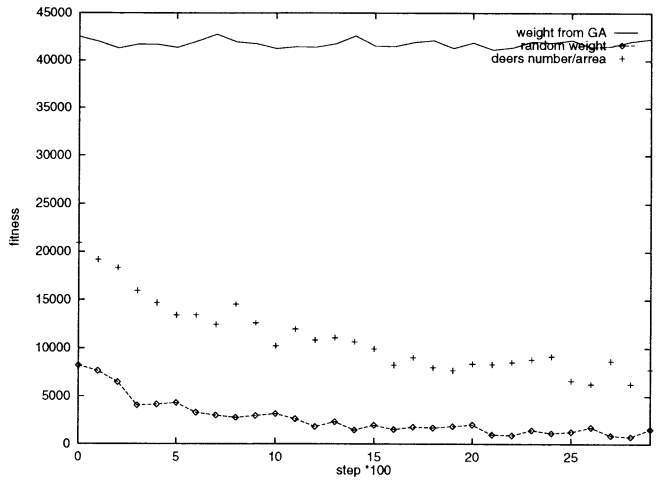


Figure 3: fitness in three cases

4 conclusion

We can obtain the suitable decision making function which describes Sika deers' actions reasonably.

We continue the examination to obtain decision making function more likely to real Sika deers.

The future, we will predict Sika deers' actions in order to manage and preserve the environment.

And we will apply this method which is used here another problem using GIS data.

References

[1] Steave. Railsback, *Emergent Thoughts on Individual-based Fish and Wildlife Modeling*,http://www.santafe.edu/projects/swarm/swarmfest99/program.html#Railsback

[2] Tim.Kohler, *Agent-Based Modeling of Anasazi Village Formation in the Northern American Southwest*, http://www.santafe.edu/projects/swarm/users/Misc/Pages/Village/paper.html

[3] C.A. Abbott, M.W. Berry, and J.C.Dempsey, E.J. Comiskey, L.J. Gross, and H.-K. Luh , *Computational Models of White-Tailed Deer in the Florida Everglades*, http://www.cs.utk.edu/ berry/deer/deer.html

[4] R. Aspinall, *Use of geographic information systems for interpreting land-use policy and modelling effects of land-use change*, Landscape ecology and geographic information systems, 1993

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Complex control approach for Sumo robot

Masanori Sugisaka and Syouichi Mishiro
Department of Electrical and Electronic Engineering
Oita University, 700 Dannoharu, Oita 870-1192, Japan
Tel: +81-97-5547831; Fax: +81-97-5547841
msugi@cc.oita-u.ac.jp smishiro@cc.oita-u.ac.jp

Abstract

The Sumo robot is adopted in the field of technology education actively by the Ministry of Education, Science, Sports, and Culture. The robot Sumo contest was carried out since 1988 in Japan. Now it attracts a strong interest around whole country.

There are two ways to attend the contest. The first one is the " independent class ". In this game, an autonomous robot is used. The robot usually is equipped with a one-chip microcomputer. The second one is the " radio-control class ". In this game, the robots are controlled by wireless control equipment.

The autonomous robot wins the game when it defeat opponent twice in three games. Effective decisive factors are, 1) to push out the opponent robot to outside of field, 2) to make the opponent robot stop etc. The contest rule consists of two. The dimension of robot to attend the contest is limited to 20cm(length) × 20cm(width). In addition, the weight of robot can not be over 3kg.

In this paper, The control method of the Sumo robot is investigated based on the experiments. The control used in the Sumo robot is now open loop. We consider a method how to control it's own behavior according to opponent movement.

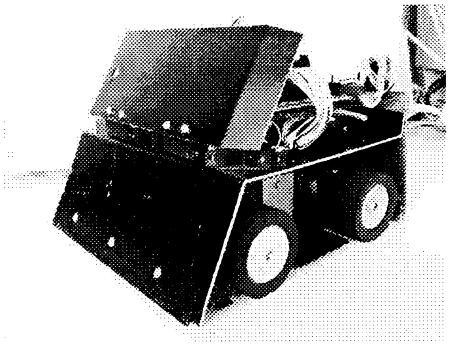


Fig.1 Whole image

1. The standard of Sumo robot

The standard of Sumo robot which we made is shown in table1,Fig.1 ,Fig.2and Fig.3. [1][2][3]

Size	Width 19cm, Depth 20cm, Height 15cm
Weight	2900g
Body	Aluminum board, Iron board
Motor	DC motor with gear
Drive method	4wheels separate drive
floor line sensor	4photo micro sensors
Object sensor	4 photo sensors
CPU	H8-3048F (Hitachi Japan)
Programming language	C, assembly
Battery	Ni-Cd battery
Motor drive voltage	DC14.4V
Others	Floor vacuum pomp flash ROM

Table1. Standard of Sumo robot

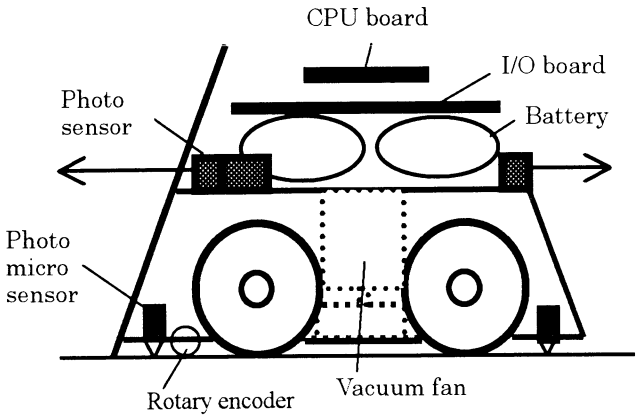


Fig.2 Side view

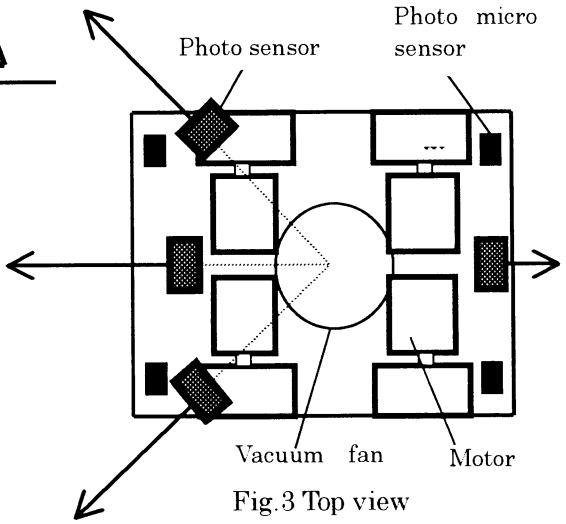


Fig.3 Top view

2. Point of control

2.1 Control flow

The control flow of the robot is shown as Fig.4. The point of control of this Sumo robot is that the Perception of circumference of the match area is the highest priority although it is important that the robot discovers the opponent robot early and bumps against it and pushes opponent out of match area. Another point is the action of robot when two object sensors perceived opponent simultaneous. Running a special routine rather than run general control program could solve the problem of the oscillation action and power down. And we made the robot to act based on the intellectual decision. The robot compared its own power and opponent power when become to push each other. Then, the robot starts the following act.

2.2 The action correspond to perception of the line sensor

In order to keep the robot from rushing out of the circumference of the match area that is indicated by a white line with the width of 5cm, We equipped line sensors at the right and left of front, right and left of the rear on base of robot. The action of the robot corresponds to the perception of the line sensor is shown as table2.

Position of active floor sensor	**action of the robot**
Front left	stop—>back—>turn left —>forward
Front right	stop—>back—>turn right —>forward
Rear right	stop—>turn left —>forward
Rear left	stop—>turn right —>forward

Table2. Action corresponding to perception of the line sensor

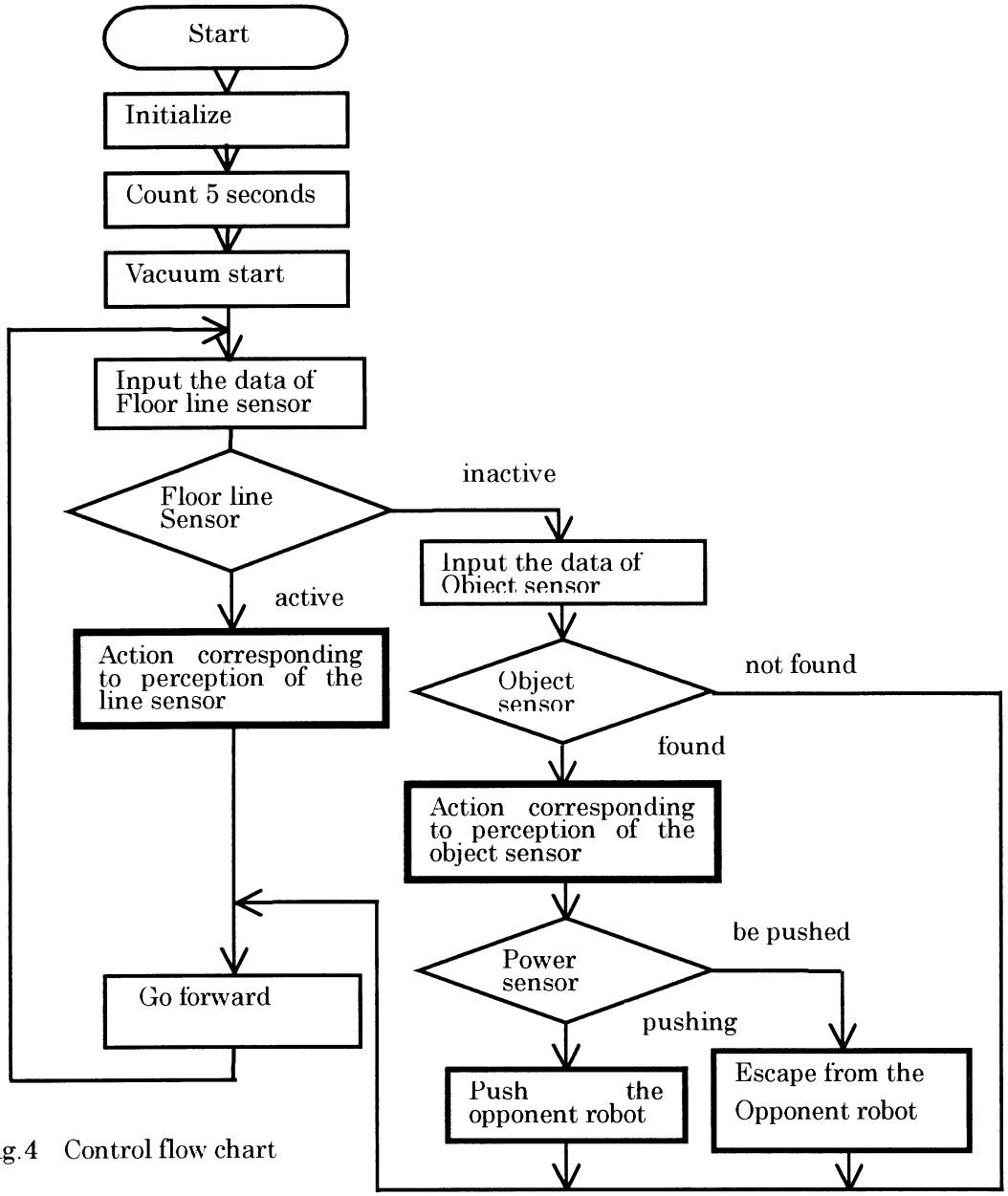


Fig.4 Control flow chart

2.3 The action correspond to perception of the object sensor

We also used three photo sensors in front of the robot and one in the rear. By these sensors, the robot finds the opponent robot and bump against it. The action of the robot corresponds to the perception of the sensor is shown in table3.

Position and direction of object sensor	**Action of the robot**
Front right 45°	turn left—>forward
Front center	Forward
Front left 45°	Turn right—>forward
Rear center	Turn 180° —>forward

Table3. Action corresponding to perception of the object sensor

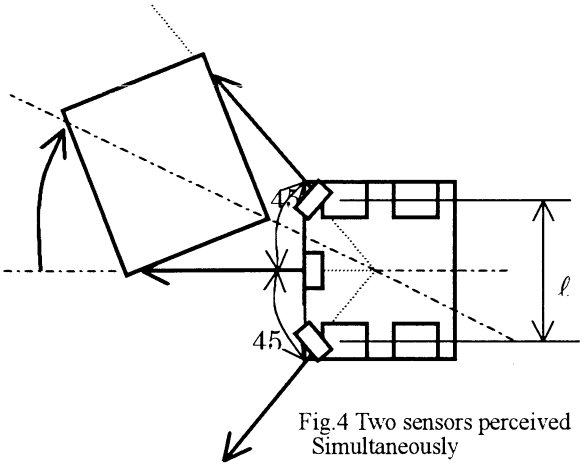


Fig.4 Two sensors perceived Simultaneously

When two photo sensors which was equipped in the center of front and right or left of front perceived opponent simultaneously such the case shown in Fig.4, unstable oscillation would occur., When two object sensors perceived opponent simultaneously, the robot go forward after the local turns of 45/2 degree, that is 22.5 degree. Local corner turn speed ω [rad/s] and turn time t[s] of the robot are showed as the next formula.

$$\omega = 2\pi \cdot \frac{1}{\pi\ell} \cdot \frac{n\pi d}{60} = \frac{n\pi d}{30\ell}[rad/s]$$

$$t = \frac{\theta}{\omega} = \frac{30\ \theta\ \ell}{n\ \pi\ d}[s]$$

n : rotation times of the wheel [rpm]

d : diameter of the wheel [m]

θ : rotation angle [rad]

ℓ : width of wheel to wheel [m]

2.4 The action when two robots push each other

The rotary encoder senses the movement of robot. If the robot slip and back when the instruction of CPU is go forward. The rotary encoder senses it. And the robot turns and escapes from the opponent robot.

3. The action experiment

After the preparation was ready, we conducted Sumo robot game in real. We found that the basic actions, such as to find the opponent robot and bump against it and to keep itself from rushing out circumference of the match area, can be done well. And the robot escapes from the opponent robot when the robot is pushed and it backed.

4. Conclusions

As the point of control of this Sumo robot, the perception of the circumference of the match area is the highest priority. It is important that the robot discovers the opponent robot early and bumps against it and pushes opponent out of match area. The robot obtained the satisfactory action by this point of view. Another point is the action of robot when two object sensors perceived opponent simultaneous. Running a special routine rather than run general control program could solve the problem of the oscillation and power down. It is computed from the local turn speed and the perception area of the sensor. The robot compared its own power and opponent power when become to push each other. Then, the robot starts the following act.

We want to make the robot to act based on the intellectual decision. From now on we consider a method how to control it's own behavior according to opponent to opponent movement. [4]

5. Reference

[1] Nishiyama ichiro: "How to design independent Sumo Robot 1", P.54 -59 , Robocon No.2,Ohmsha,1999

[2] Nishiyama ichiro: "How to design independent Sumo Robot 2", P.64 -73 , Robocon No.3,Ohmsha,1999

[3] Nishiyama ichiro: "How to design independent Sumo Robot 5", P.70 -77 , Robocon No.6,Ohmsha,1999

[4] Mashanori .Sugisaka: "Introduction, system theory and control ",P.166-180, Nikkan industry ,1997

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Applications of Conscious Models to Human-Machine Cooperative Systems

*Kouichi Nakazato, **Eiichi Bamba

*Ishikawajima-Harima Heavy Industries co.,Ltd.
2-12,Honmachi,4-chyoume,Chuo-ku,Osaka 541-0053,Japan
E-mail:<kouichi_nakazato@ihi.co.jp>

**Department of Mechanical Engineering, Faculty of Science and Technology, Kinki University
3-4-1 Kowakae, Higashiosaka, Osaka 577-8502,Japan
E-mail:<bamba@mec.kindai.ac.jp>

Abstract : Many cases of manufacturing tasks, the human operators need to adapt for the change of environments. In Agile Manufacturing Systems, the changes are more complex and dynamic. This paper trys to make Human Factor Models correspond the tasks changing dynamically.

We have already demonstrated in other paper presented by ourselves that Human Factor Model is made up of three subsystems found on Control System Constructual Theory. We try to adapt this theory for "Obstacle Avoiding Problems".

For the first step, this paper presents Human Factor Model which can find the way of avoiding by using its knowledge. We teach the avoiding way beforehand for the Model and after that, the Model can output the way of avoiding according to the knowledge. This Model can avoid the obstacle and reach the goal point.

As stated above, this Model decides its action by using its knowledge teached beforehand. So the case which have never teached, the outputs often don't fit. For the next step, we give the Model the faculty of studying. If the Model fails the avoiding, trys another way and successes avoiding, the Model studys the way. We call this Model as "Contextual Model."

Key word : Human Factor Model , Contextual Model , Obstacle Avoiding Problem , Hierarchical Neural Networks

1,Introduction

Traditionary working strategies based on Industrial robots are named "Sequential Models" in this paper. But, recent works in manufactures demand to product complicable and multi sort dynamically and swiftly. In case of those strategies, human and machine cooperations are indispensable.

This paper verifies to make human and machine cooperative models realizable by introducing conscious models related to self organized functions of human models taken in human factors. Futher, this paper defines Contextual Models endowed the above models with skilled functions, and presents models understandable like humen.

Recently, the general idea of "Contextual Model" gets much attention. This paper suggests Contextual Model takes into consideration the Human Cognitive Factor. We've demonstrated that Human Factor Models are made up of three subsystems found on Control System Constructual Theory. To prove it's right, we trys to adapt this theory for "Obstacle Avoiding Problems".

For the first step, we try to make up Model which outputs the way of avoiding by using its knowledge, and avoids obstacle of wall. The Model is that, if information of its circumference is inputted, Hierarchical Neural networks outputs the way of avoiding. This paper indicates the results of avoiding obstacle.

From the result, it is found that Contextual Model must have faculty of studying. If we can give the Model the faculty of studying , the Model becomes more skillful through the experience of work. This paper defines such Model as skillfulness of Contextual Model.

2,Theory

(1)Constructions of Human Factor Models

It is already demonstrated in other paper presented by ourselves that Human Factor Models are made up of three subsystems found on Control System Constructual Theory, i.e. controllable and observable systems (C/OS), controllable and unobservable systems (C/UOS), and uncontrollable and observable systems (UC/OS).

C/OS. means subsystems to accept and carryout the pointed tasks. C/UOS. accept the pointed tasks and collate with the outputs of UC/OS. UC/OS can't accept the pointed tasks and affect outputs of C/OS.

Human Factor Models integrated by the above subsystems operate to excute the tasks by C/OS. and accept the stimuli by C/UOS.when C/UOS. Conclude to be impossible tasks continuing to work by C./OS., then

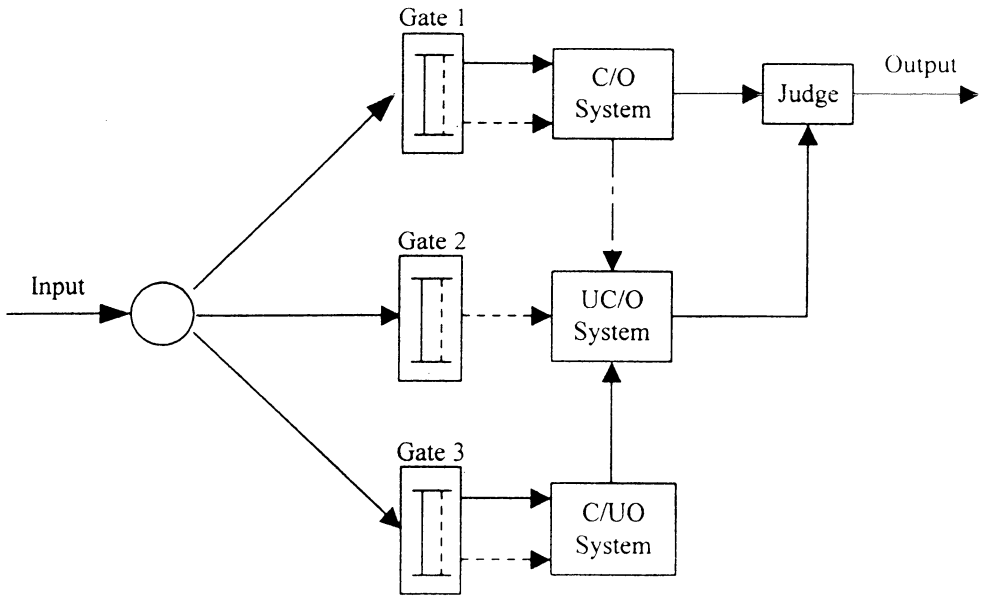


Fig.1 Constructions of Human Factor Models

those subsystems inquire settlling method of UC/OS. Fig.1. is the construction of Human Factor Model we have proposed.

(2)Realizable Example of Human Factor Model

By using the above Human Factor Models ensured to reflect human factor mechanisms theoretically, this paper applies to the obstacle avoiding problem with autonomy vehicle naigations, and gives computer simulations.

Contents of the task are assumes that the autonomy vehicle navigates from the start point, A to the final point, Bon the replete surface with obstacles. :For carring out those tasks, C/OS. command the autonomy vehicle to progress from A to B. If any, C/UOS. detect the obstacles on the way. C/UOS. determine the obstacles by C/UOS.

In this paper, the avoiding methods of the obstacles are previously trained by hierarchial type neural network, and outputs of this neural network are so framed that predictable to be optimal corresponding to the shapes of the obstacles. Simulational experiments using those systems attain our purpose at the first stage. Fig.2. shows the Realizable Example of Human Factor Models.

Fig.3. shows the Flowchart of Human Factor Models. If the Model gets task and can go straight to the final point, the C/O system is carryed out. If the Model can't go straight to the final point, C/UO system and UC/O system is carryed out. If the output of UC/O system does not fit, the Model carrys out the Random Walk.

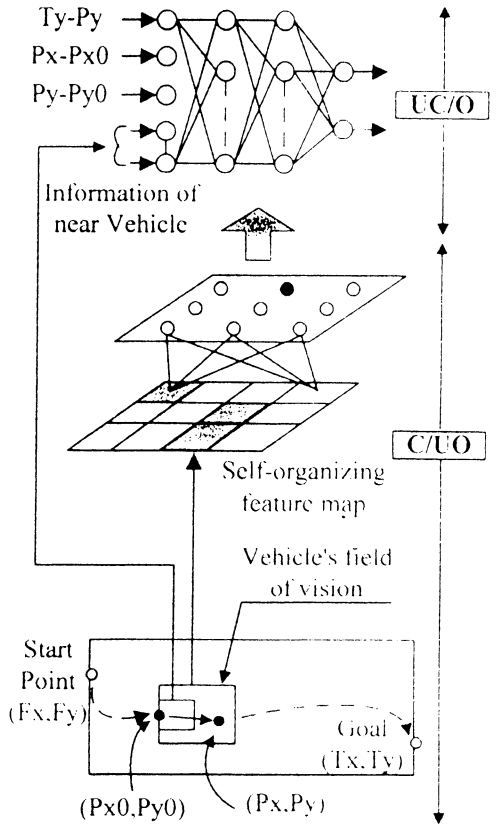


Fig.2 Realizable Example of Human Factor Model

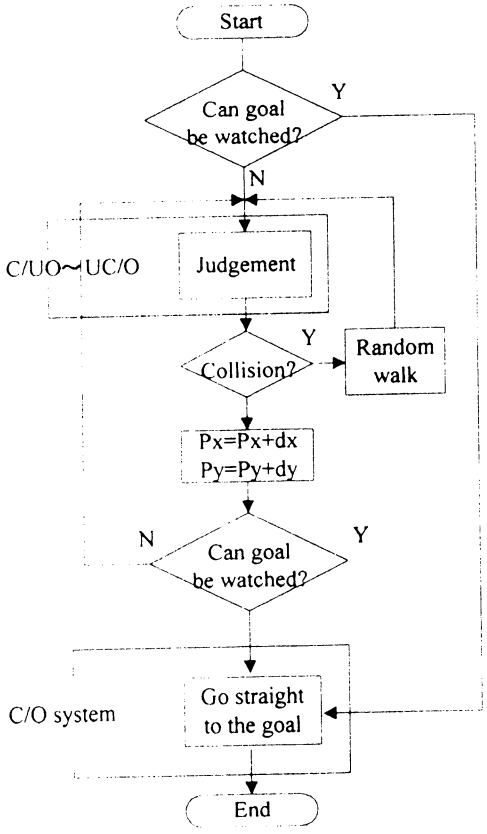


Fig.3. Flowchart of Human Factor Models

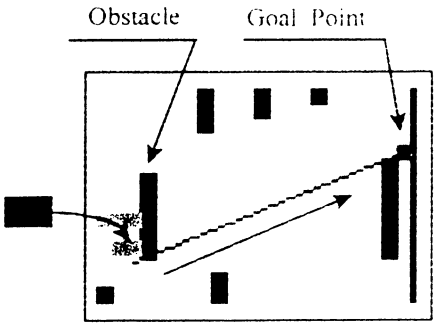


Fig.4-1 C/O System

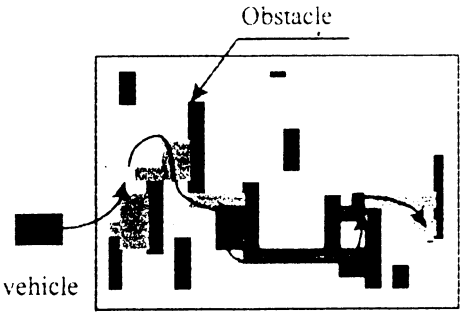


Fig.4-2 C/UO→ UC/O System

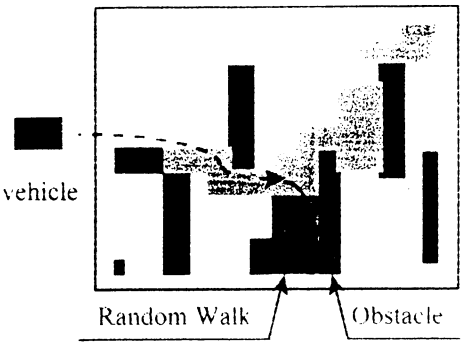


Fig.4-3 Random Walk

4,Result

Fig.4.1～4.3 shows the results of Obstacle Avoiding Problems by using above Human Factor Models.

Fig.4.1 shows "C/O System". The vehicle go straight to the final point when it finds that no obstacle exists between vehicle and final point. This action is equal to the given tasks.

Fig.4.2 shows "C/UO System → UC/O System". If obstacle exists between vehicle and final point, the information of vehicle's circumference is inputted in C/UO System. C/UO System extracts its characteristics and selects Hierarchical Neural Networks. As the Hierarchical Neural Networks is studied beforehand, UC/O System outputs the way of avoiding. The arrow line shows the action of this Systems. It is found that the vehicle avoide the obstacle of wall.

Fig.4.3 shows the case that C/UO System and UC/O Sytem can't find the fitting way of avoiding. As vehicle can't find the way, it loses its way and acts random walk. Vehicle finds its way after random walk for some times and carry out above C/UO System and UC/O System.

5.discussions

From the results of Computer Simulation, following are found.

(1) Above Human Factor Model can solve the "Obstacle Avoiding Problem". The Model judges the condition and how to avoid the obstacle. It is found that the Model showed in Fig.1 is right.

(2) In the circumstances of dynamical and confusable works, it is impossible that the models gain all of the working systems are necessary to have faculties acquiring functions. In this paper, those faculties acquiring functions are defined as Skillful functions. Skillful functions are made up learning and reasoning functions.Learning functions master executing methods by experiencing some precedences, and reasoning functions recognize similarity of works on knowledges of learning functions. UC/OS. in Human Factor Models equipped with those two functions make skillness incleasible, and variances of situations adaptables. Human Factor Models annexed skillful functions are defined as Contextual Models in this paper.

(3) In studying, one of the difficult, but important factor is analogy. When we can't find good way about the problem, then we look for similar pattern. Similarly, the Model should look for similar pattern which can use for the case.

(4) This paper tries to solve obstacle avoiding problem and the obstacle is wall. For the next step, we must try to adapt this Model for avoiding of moving things. When the output doesn't fit for the condition, the Model stops to find another way. If the object is moving and the Model stops, they may collide with each other.

6.Conclusions

This paper presents not traditionary manufacturing models behavioring time sequentially but Human Factor Models correspondable pliantly to dynamic and inticate variances of tasks contents. Presented models are based on conscious models assured by Control Systems Structual Theory and models annexed skillful functions to Human Factor Models, Contextual Models are presented. By adapting Contextual Model for agile Manufacturing Systems, the usefulness of Contextual Models are made obvious by applying to tasks carried out under the circumstances of existing obstacle and demonstrated by computer simulations.

7.Acknowledgments

This work is supported by Grant-in-Aid for Scientific Research(B)(2)国11695062of Japan Society for the Promotion of Science.

8.References

[1] Bamba,E., Nakajima,W .,Liqin,R(1997).
A Participation of Mental Factors in Agile Manufacturing Systems. In Salvendy,G, Smith,M. & Koubek,R.(Eds.):
Design of Computing Systems: Cognitive Considerations (Vol.1), Proc. 7th Int. Conference of Human-Computer Interaction (HCI International '97, San Francisco, USA, August24-29,1997),pp.339-343.Amsterdam:Elsevier.

[2] Bamba,E., Nakazato,K., Fujita,T.(1999).
Human Cognitive Model for Organization Design suited to Task in Agile Manufacturing. In Bullinger,H.J. & Ziegler,J.(Eds.):Communication Cooperation, and Application Design (Vol.2), Proc. 8th Int.Conference on Human-Computer Interaction (HCI International '99, Munich, Germany, Augusut 22-26,1999),pp.1231-1235. London : LAWRENCE ERLBAUM ASSOCIATES PUBLISHERS.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Nonlinear Control of Snake Robot

P. Prautsch T. Mita
Department of Control and Systems Engineering
Tokyo Institute of Technology
Tokyo, 152-0033

Abstract

Consider a snake robot that consists of n rigid links and $n-1$ actuated joints and free wheels attached to center of mass of each link. We present a head position and tracking control based on dynamical model. In order to achieve an exact tracking of a desired path, we remove wheels from the first link and using nullspace motion we steer the head of a kinematic model along the desired trajectory.

Keywords: nonlinear control, nonholonomic system, snake, nullspace motion

1 Introduction

Snake robots, modeling a creature that is admired for being able to move in quite a special way, have been of interest in the last several years. Considering them just as mobile redundant manipulators leads to design of snake robots that do not use wheels and and their traveling velocity is quite low [1, 2]. Starting with Hirose [3], the difference in tangential and normal friction of the snake's belly and the ground is also approximated using wheels. That provides higher motion efficiency and thus increases the robot's velocity. Moreover, using wheels also yields a class of nonholonomic systems, which are of particular theoretical interest in robotic and control literature [4].

However, position control of such machine has proven to be difficult [3] or has not been considered [5]. In this study, we will review our previous result of snake robot head position control and then extend it for tracking a given path. The obstacle in tracking is that for a wheeled robot a straight motion forward is possible without a snake-like locomotion. However, in that case the machine becomes a trailer-like mobile robot.

For the sake of brevity, we will use the following abbreviation wherever the meaning is clear

$$\mathrm{C}\theta = \cos\theta,\ \mathrm{S}\theta = \sin\theta,\ \ \mathrm{C}_i = \cos\theta_i,\ \mathrm{S}_i = \sin\theta_i,$$
$$\mathrm{C}_{ij} = \cos(\theta_i - \theta_j),\ \ \mathrm{S}_{ij} = \sin(\theta_i - \theta_j).$$

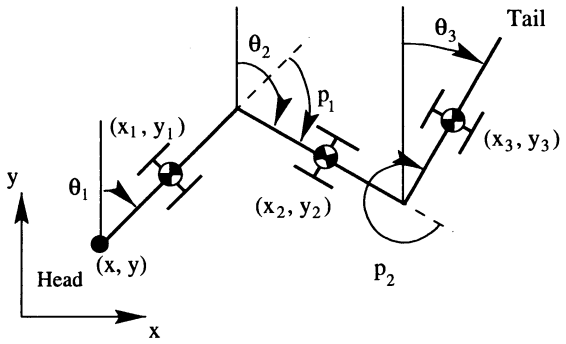


Figure 1: Snake robot

2 Robot model

We consider a mobile robot that consists of n rigid links and $(n-1)$ actuated joints as shown in Fig. 1. The length $2l$ and weight m are the same for each link and the distribution of weight is assumed to be uniform. The the center of mass (CM) of each link is located in the middle of the link.

Orientation of each link and location of corresponding CM will be denoted as θ_i and (x_i, y_i), $i = 1 \sim n$, respectively. The absolute angle θ_i is measured clockwise from the y axis. The realative angles $p_i = \theta_{i+1} - \theta_i$ $(i = 1 \sim n-1)$ are controlled by input torque $u \in R^{n-1}$. Furthermore, we denote (x, y) as the position of the head. In the presented model, a fixed axis with two passive wheels is attached to the CM of each link and prevents thus the CM from sliding sideways.

Then, as can be seen from Fig. 1, the position and velocity of each CM are given by

$$\begin{aligned} x_i &= x + 2l\Sigma_{j=1}^{i-1}\mathrm{S}\theta_j + l\mathrm{S}\theta_i \\ y_i &= y + 2l\Sigma_{j=1}^{i-1}\mathrm{C}\theta_j + l\mathrm{C}\theta_i \end{aligned} \tag{1}$$

and

$$\begin{aligned} \dot{x}_i &= \dot{x} + 2l\Sigma_{j=1}^{i-1}\dot{\theta}_j\mathrm{C}\theta_j + l\dot{\theta}_i\mathrm{C}\theta_i \\ \dot{y}_i &= \dot{y} - 2l\Sigma_{j=1}^{i-1}\dot{\theta}_j\mathrm{S}\theta_j - l\dot{\theta}_i\mathrm{S}\theta_i. \end{aligned} \tag{2}$$

From the assumption that the wheels cannot slip sideways follows a velocity constraint for each link

$$\dot{x}_i \mathrm{C}\theta_i - \dot{y}_i \mathrm{S}\theta_i = 0 \quad : i = 1 \sim n \tag{3}$$

and by substituting (2) into (3) we get

$$l\dot{\theta}_i + 2l\Sigma_{j=1}^{i-1}\dot{\theta}_j \mathrm{C}_{ij} + \mathrm{C}\theta_i \dot{x} - \mathrm{S}\theta_i \dot{y} = 0 \ : i = 1 \sim n. \tag{4}$$

Rewrite these velocity constraints in a matrix form as

$$F_A \dot{\theta} = F_B \dot{w} : \quad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ : \\ \theta_n \end{bmatrix}, w = \begin{bmatrix} x \\ y \end{bmatrix}, \tag{5}$$

where $FA \in R^{n \times n}$, $FB \in R^{n \times 2}$. Equation (5) can be rewriten as

$$\dot{\theta} = F(\theta)\dot{w} : \quad F = F_A^{-1} F_B. \tag{6}$$

Further, multiplying (5) by F_A^{-1} yields

$$[I_n, -F(\theta)] \begin{bmatrix} \dot{\theta} \\ \dot{w} \end{bmatrix} = A(q)\dot{q} = 0 \tag{7}$$

as its Pfaffian form [4], where $q = (\theta, w)^T \in R^{n+2}$ denotes the generalized coordinate. Equations (5) are not integrable, so we deal with a nonholonomic system. For

$$p = E^T \theta \quad : p = (p_1, p_2, ..., p_{n-1})^T, \ E^T \in R^{n-1 \times n} \tag{8}$$

holds, the velocity constraint in terms of the relative angle is given by

$$\dot{p} = E^T F \dot{w}. \tag{9}$$

Before proceeding to dynamic models we shall recall from [6] that Kinematic singularities occur when

Lemma 1 *Matrix $E^T F(n-1 \times 2)$ loses rank if the robot becomes straight, i.e., $p_i = 0 \pmod{\pi}$ or it constitutes an arc of a ring, i.e. $p_1 = p_2 = ... = p_{n-1}$.*

We will use two dynamic models. The first is derived by incorporating the Pfaffian constraints (7) on equations of motion of a robot without any velocity constraints as follows

$$\begin{aligned} &M(\theta) \begin{bmatrix} \ddot{\theta} \\ \ddot{w} \end{bmatrix} + \begin{bmatrix} C_1(\dot{\theta}, \theta) \\ C_2(\dot{\theta}, \theta) \end{bmatrix} \dot{\theta} + D(\theta) \begin{bmatrix} \dot{\theta} \\ \dot{w} \end{bmatrix} \\ &+U(\theta) + A(\theta)^T \lambda - \begin{bmatrix} Eu \\ 0 \end{bmatrix} = 0, \end{aligned} \tag{10}$$

where the matrices $M(\theta)$, $\begin{bmatrix} C_1(\dot{\theta}, \theta) \\ C_2(\dot{\theta}, \theta) \end{bmatrix}$, $D(\theta)$ and $U(\theta)$ stand for the inertial, centrifugal, dissipative and gravity term, and λ is a Lagrange multiplier. Canceling λ in (10) yields

$$\tilde{M}(\theta)\ddot{w} + \tilde{C}(\theta, \dot{\theta})\dot{w} + \tilde{D}(\theta)\dot{w} + \tilde{U}(\theta) = F^T E u, \tag{11}$$

where

$$\begin{aligned} &\tilde{M} = (F^T, I_2) M \begin{bmatrix} F \\ I_2 \end{bmatrix} > 0, \\ &\tilde{D} = (F^T, I_2) D \begin{bmatrix} F \\ I_2 \end{bmatrix} > 0, \ \tilde{U} = (F^T, I_2) U, \\ &\tilde{C} = (F^T M_{11} + M_{12}^T)\dot{F} + F^T C_1 F + C_2 F. \end{aligned}$$

Second model is obtained using a transformation matrix $G(\theta)$ and relation

$$G\dot{w} = \begin{bmatrix} l\dot{\theta}_1 \\ \dot{\eta}_1 \end{bmatrix}, \tag{12}$$

where

$$G = \begin{bmatrix} -\mathrm{C}\theta_1 & \mathrm{S}\theta_1 \\ \mathrm{S}\theta_1 & \mathrm{C}\theta_1 \end{bmatrix} = G^T = G^{-1} \tag{13}$$

and $\dot{\eta}$ is the tangential velocity of the head $\dot{\eta}_1 = \dot{x}\mathrm{S}\theta_1 + \dot{y}\mathrm{C}\theta_1$. Transforming the coordinates w in (11) yields

$$\bar{M} \begin{bmatrix} l\ddot{\theta}_1 \\ \ddot{\eta}_1 \end{bmatrix} + \bar{C} \begin{bmatrix} l\dot{\theta}_1 \\ \dot{\eta}_1 \end{bmatrix} + \bar{D} \begin{bmatrix} l\dot{\theta}_1 \\ \dot{\eta}_1 \end{bmatrix} + \bar{U} = GF^T E u. \tag{14}$$

3 Position control

Position control of the dynamical snake employs Lyapunov method together with model (11). Proposing Lyapunov function candidate

$$V = \frac{1}{2}\dot{w}^T \tilde{M} \dot{w} + \frac{1}{2}(w - r)^T B_0 (w - r), \tag{15}$$

with reference position $r \in R^2$ and design parameter $B_0 > 0$ and control

$$u = -A_0 E^T F \dot{w} - (F^T E)^{\dagger}(B_0(w - r) - \tilde{U}) \tag{16}$$

where $A_0 \geq 0$ is another design parameter we get

$$\dot{V} = -\dot{w}^T (\tilde{D} + F^T E A_0 E^T F)\dot{w} < 0.$$

We have used skew symmetric property of matrices in (11), see [6] for proof. Thus position of the robot head asymptotically converges to the reference position.

In order to reduce the requirements on torque inputs, the orientation of the head can be constrained to a serpenoid function

$$\theta = \omega \sin(k\eta), \tag{17}$$

where η is a distance along the curve from some reference point. This curve was proposed in [3] as the

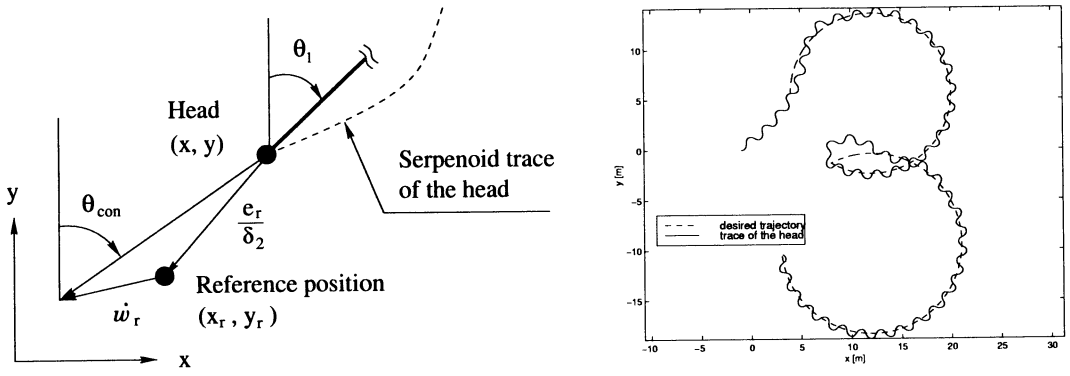


Figure 2: Tracking using serpenoid curve

best description of the natural snake motion. The set of n velocity constraints (4) then becomes intergrable (holonomic) and the snake robot with $(n+2)$ degrees of freedom is represented by system with one degree of freedom on an invariant manifold [6].

To achieve that, we use zeroing method and write the control constraint and distance control as

$$\begin{aligned} F_1 &= l\theta_1 - l\omega\sin(k\eta_1) = 0 \\ F_2 &= \eta_1 - \gamma = 0, \end{aligned} \tag{18}$$

where γ is a reference distance along the curve. The dynamics of convergence to these constraints is determined by

$$\begin{aligned} \ddot{F}_1 + \alpha\dot{F}_1 + \beta F_1 &= 0 \\ \ddot{\eta}_1 + \vartheta\dot{\eta}_1 + \xi(\eta_1 - \gamma) &= 0 \end{aligned} \tag{19}$$

where $\alpha > 0$, $\beta > 0$, $\vartheta > 0$, $\xi > 0$ are constant coefficients. That yields the controk u as

$$u = (GF^TE)^\dagger \left(\bar{C}\begin{bmatrix} l\dot{\theta}_1 \\ \dot{\eta}_1 \end{bmatrix} + \bar{D}\begin{bmatrix} l\dot{\theta}_1 \\ \dot{\eta}_1 \end{bmatrix} + \bar{U} + \bar{M}P^{-1}K\right), \tag{20}$$

where

$$K = \begin{bmatrix} -l\omega k^2\sin(k\eta_1)\dot{\eta}_1^2 - \alpha\dot{F}_1 - \beta F_1 \\ -\vartheta\dot{\eta}_1 - \xi(\eta_1 - \gamma) \end{bmatrix}.$$

Then the creeping motion of the snake robot will satisfy (18), for details see [6]. Parameters ω and k of the serpenoid curve are to be adjusted in order to achieve the least torque.

4 Tracking

Tracking means following a given a given path with given velocity and orientation. Since the snake moves forward by changing its inertial shape, we have to relax the condition of orientation and require only velocity control at given position.

Now, we recall from [6] that

Lemma 2 *When $\dot{w}$=v=constant, $p_i = 0\,(mod\ \pi)$ becomes a stable equilibrium of $\dot{\theta} = F(\theta)\dot{w}$.*

This zero dynamics determining the robot shape for given constant velocity becomes straight and thus singular applies also for dynamic case (11) even if a penalty function is included into Lyapunov function candidate below

$$V = \frac{1}{2}\frac{(\dot{w} - \dot{r})^T B_0(\dot{w} - \dot{r})}{\theta^T R\theta},$$

where $R = EE^T$ and thus $\theta^T R\theta = p^T p$ should prevent the body to become straight.

Then, we will use the serpenoid curve, replace the distance control in (18) by a velocity control and adjust the orientation and velocity according to the left part of Fig. 2 as

$$\begin{aligned} F_1 &= l\theta_1 - l\omega\sin(k\eta_1) + \theta_{con} = 0 \\ F_2 &= \dot{\eta}_1 - (\dot{w}_d + \delta_1 e_d) = 0, \end{aligned} \tag{21}$$

where δ_1 is a positive control parameter, and $e_d = (x_d - x)\cos(\theta_{con}) + (y_d - y)\sin(\theta_{con})$ is a distance from the current position to the desired one in the direction of the desired movement.

Even though the robot's head is tracking a desired trajectory in average, the head can never follow the path exactly. This is caused by full column rank of matrix FA in (5). In order to be able to follow the desired path precisely, we have to give the system more freedom. That corresponds to a real snake whose head is lifted and does not touch the ground.

From now on, we will consider only kinematic control. By attaching the wheels to link i, $i = (2 \sim n)$ (5)

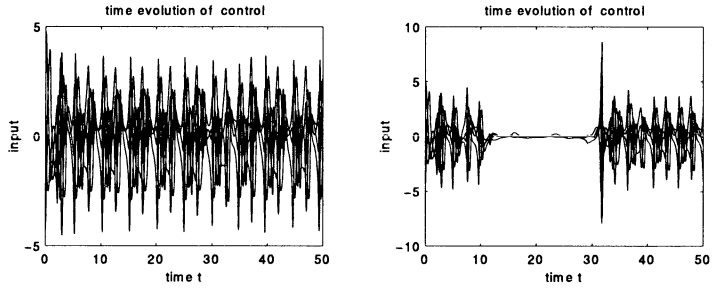


Figure 3: Input of kinematic motion for maximizing $g_1(\theta)$

becomes

$$FA\dot{\theta} = FB\dot{w}: \quad FA \in R^{n-1\times n},\ FB \in R^{n-1\times 2}. \tag{22}$$

Since the vector $\dot{\theta}$ can be for any control strategy expressed as

$$\dot{\theta} = FA^{\dagger}FB\dot{w} \pm Pv, \tag{23}$$

where $P = (I - FA^{\dagger}FA)$ is an orthogonal projection onto the null space of FA, and v is a continuous weighting vector [7].

The nullspace motion determined by P and v can be used to maximize some criterion such as one of

$$g_1(\theta) = 1/2 \sum_{I=2}^{n-1} p_I^2$$
$$g_2(\theta) = a/2 \det(FAFA^T) + b/2 \det(FB^T FB)$$

Here, $g_1(\theta)$ is trying to get the shape of the snake as far as possible from stable equilibrium given in Lemma 2. On the other hand, $g_2(\theta)$ maximizes a manipulability-like index. Then, v can be computed as

$$v = \bigtriangledown g(\theta). \tag{24}$$

The control $u = \dot{p}$ is then given as

$$u = E^T \left(FB^{\dagger}FA(\dot{r} - K(w - r)) + A_0(I - FA^{\dagger}FA\right) v. \tag{25}$$

Simulation result of a 5-link robot controlled to move along a straight line with constant velocity is given in Fig. 3. We were maximizing the function $g_1(\theta)$ and it is obvious that even though the control (25) steers the robot away from the stable equilibrium, the behavior of the snake can not be predicted. The left part shows control input for $K = 2$, $A_0 = 1.8$. By slightly increasing the value of parameter A_0 to 2 we obtain a very different behavior, where the robot takes singular posture of a while, right part of Fig. 3. As for criterion $g_2(\theta)$, we have noticed that the matrix important for steering the robot away from singular postures is the matrix FB which in accordance with proof of Lemma 1. However, maximizing singularity yields a zig-zag pattern. That does not correspond to the common idea of a snake-like locomotion.

5 Conclusion

In this paper, we have shown a basic idea to position control and tracking of a snake robot. In order to achieve a predictable behavior, we would need to move the motion on some invariable manifold. However, e.g. extended Jacobian method yields an algorithmic singularity for a snake with free first link. This topic is still a matter of further investigation.

Acknowledgements

This research has been supported by the grant Titech COE Super-Mechano-System.

References

[1] Y. Shan, Y. Koren,"Design and Motion Planning of a Mechanical Snake",*IEEE Trans. Syst., Man, Cybern.*, Vol. 23, pp. 1091-1100, 1993.

[2] M. Nilsson, "Snake Robot Free Climbing", *IEEE Control Syst. Magazine*, Vol. 18, pp. 21-26, 1998.

[3] S. Hirose, *Biologically Inspired Robots: Snake-Like Locomotors and Manipulators*,Oxford University Press, 1993.

[4] R. M. Murray et al., *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 1993.

[5] J. Ostrowski, J. Burdick, "The Geometric Mechanics of Undulatory Robotic Locomotion", *Int. J. of Robotics Research*, Vol. 17, pp. 683-701, 1998.

[6] P. Prautsch et al., "Analysis and Control of a Gait of Snake Robot", *IEEJ Trans. D*, Vol. 120, 2000.

[7] T. Shamir, K. Yomdin, "Repeatability of Redundant Manipulators: Mathematical Solution of the Problem", *IEEE Trans. Aut. Contr.*, Vol. 33, 1988.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Stabilizing Control of Passive Biped Robot and Its Application to Active Walking

Fumihiko Asano
Department of Control Engineering
Tokyo Institute of Technology
2-12-1 Oh-okayama,
Meguro-ku,
Tokyo 152-8552, JAPAN
asano@ctrl.titech.ac.jp

Masaki Yamakita
Department of Control and Systems Engineering
Tokyo Institute of Technology
2-12-1 Oh-okayama,
Meguro-ku,
Tokyo 152-8552, JAPAN
yamakita@ctrl.titech.ac.jp

Katsuhisa Furuta
Department of Mechanical and Environmental Informatics
Tokyo Institute of Technology
2-12-1 Oh-okayama,
Meguro-ku,
Tokyo 152-8552, JAPAN
furuta@mei.titech.ac.jp

Abstract

This paper propose a control law of simple biped walking robot based on passive dynamic walking. We show that the robot can also walk on the level in the presence of "virtual gravity field" and tranceform the gravity effect to torque of actuators without any approximations. Then active walking with small torque can be realized, however, passive or virtual passive walking motion is very sensitive to initial condition and disturbances. Thus we propose a control law to make the walking system robust and the control does not injure "virtual passivity". According to above method, safety and small-torque walking can be realized. The validity of control law is investigated by computer simulation.

Key words: Robotics, biped robot, legged locomotion, passive dynamics.

1 Introduction

The study of bipedal walking in the framework of humanoid robot is resent active research area. The gait design is most important part of walking robot control, but in most of past researches they imitate human gait intuitively without any solid reason which insures some optimality, especially energy consumption.

It is clear that the human gait depends on the geometry and internal properties of human body. Since human can walk a long distance with small energy supply, the gait takes advantage of the gravity effect as "passive" walk. Based on this observation, energy-effective gait of active walking robot on the level can be obtained based on that of passive walk; it is referred to as "dynamics based control".

McGeer designed several unpowered biped robots and studied their gravity-induced passive motion on inclined planes[3]. He demonstrated that the prototype can attain a stable steady periodic motion and analyzed this behavior with a linearized mathematical model. INRIA studied such a passive system by means of its full nonlinear equations and verified the symmetric and chaotic motion [1]. The passive motion has a special appeal because it is *natural* and it does not require any external energy source.

In this research, we use the steady walking pattern of "virtual passive" walking robot as a model of an active walking robot which walks on the level by actuators. The stable limit cycle of virtual passive walk is, however, very sensitive to its initial condition and weak to disturbances. We defined stable or optimal gait as "continuous walking without control forces", and we propose an impact control law, "Real-time slope driven control"[4], to keep "virtual passivity" during single support phase. Above method utilizes physical characteristic of the robot effectively to walk on the level with small energy.

We investigate the validity of the control method by computer simulation and analyze its stability.

2 Passive Dynamic Walking

The model of the biped robot we consider is shown in Fig.1. In this model the legs are rigid bars without knee and foot, connected by a frictionless hinge at the hip. The motion is assumed to be restricted in the sagittal plane. The robot can walk down shallow slopes without any control forces and converges 1-periodic stable limit cycle.

During the swing (single support phase), the robot behaves like a planar double-pendulum. Its dynamical

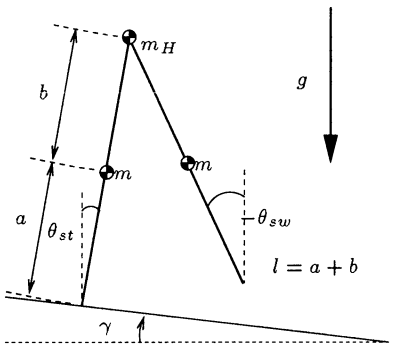


Figure 1: Model of a compass gait biped robot on a slope

equations can be derived using the well-known Euler-Lagrange approach and we get dynamic equation of the robot as follows:

$$\boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + \boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{0} \tag{1}$$

Where $\boldsymbol{\theta} = \begin{bmatrix} \theta_{st} & \theta_{sw} \end{bmatrix}^T$ is the coordinates of the configuration of the robot, $\boldsymbol{M}(\boldsymbol{\theta}) = [2\times2]$ is the inertia matrix, $\boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = [2 \times 2]$ is the coriolis matrix, $\boldsymbol{g}(\boldsymbol{\theta}) = [2 \times 1]$ is the gravity matrix. The details are omitted.

The algebraic transition equations relate the robot's states just before and just after its collision with the ground. They can be simply expressed by matrices, however, the details are omitted.

Here, we must add that compass gait cannot exist in reality, either biologically or machine. This is because the fact that the swing leg, since it does not have a knee, cannot actually clear the ground. So the foot-scuff must be omitted in computer simulation.

3 Stabilizing Control

3.1 Problem formulation

In this research, we propose a control law which does not injure passivity or "virtual passivity", so the walking robot cannot use any actuators to stabilize itself. There are, however, many reasons which make continue-walking unstable. Then we propose "Real-time slope driven control"[4].

The control system is given by:

$$\gamma_k = \gamma^* + \delta\gamma_k + \sigma w_k \tag{2}$$

and $\delta\gamma_k$ is control input. In the mode we assume the ruggedness on the slope as disturbance and it is represented as σw_k where w_k is normal Gaussian noise and σ is its weight.

3.2 Discrete time LQR optimal control

The $k+1$th state of the robot is uniquely determined by kth state of the robot and slope, so it is considered a discrete time nonlinear system as follows:

$$\boldsymbol{q}_{k+1} = \boldsymbol{F}(\boldsymbol{q}_k, \gamma_k) \tag{3}$$

where $\boldsymbol{q} = \begin{bmatrix} \boldsymbol{\theta}^T & \dot{\boldsymbol{\theta}}^T \end{bmatrix}^T$ and $\boldsymbol{q}_k$ is kth post-impact state of the robot. Especially, if the state vector $\boldsymbol{q}^*$ is on a desired limit cycle, the following equation hold.

$$\boldsymbol{q}^* = \boldsymbol{F}(\boldsymbol{q}^*, \gamma^*). \tag{4}$$

Let us perturb $\boldsymbol{q}^*$ and γ^* and derive the propagation of the perturbation. A small perturbation $\delta\boldsymbol{q}_k$ at step k manifests itself as a deviation $\delta\boldsymbol{q}_{k+1}$ at step $k+1$, while satisfying Eq.(3) as

$$\boldsymbol{q}^* + \delta\boldsymbol{q}_{k+1} = \boldsymbol{F}(\boldsymbol{q}^* + \delta\boldsymbol{q}_k, \gamma^* + \delta\gamma_k), \tag{5}$$

Linearizing Eq.(5) around $\boldsymbol{q}^*$ and γ^* and using Eq.(4), we get:

$$\delta\boldsymbol{q}_{k+1} = \left.\frac{\partial \boldsymbol{F}(\boldsymbol{q}, \gamma^*)}{\partial \boldsymbol{q}}\right|_{\boldsymbol{q}=\boldsymbol{q}^*} \delta\boldsymbol{q}_k + \left.\frac{\partial \boldsymbol{F}(\boldsymbol{q}^*, \gamma)}{\partial \gamma}\right|_{\gamma=\gamma^*} \delta\gamma_k. \tag{6}$$

Eq. (6) can be expressed as following a discrete time linear system:

$$\boldsymbol{x}_{k+1} = \boldsymbol{\Phi}\boldsymbol{x}_k + \boldsymbol{\Gamma}u_k. \tag{7}$$

Since $\boldsymbol{\Phi} = [4 \times 4]$ and $\boldsymbol{\Gamma} = [4 \times 1]$ are cannot be obtained analytically, they must be computed numerically by computer simulation. The cycle is stable and $(\boldsymbol{\Phi}, \boldsymbol{\Gamma})$ is controllable.

In order to stabilize the motion, the criterion is defined as:

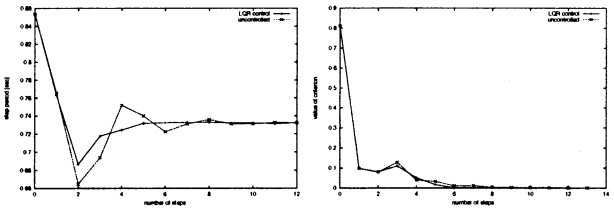
$$J := \sum_{k=0}^{\infty} (\boldsymbol{x}_k^T \boldsymbol{Q} \boldsymbol{x}_k + r u_k^2), \tag{8}$$

and we can solve LQR optimal control for the criterion function to determine u_k.

3.3 Numerical simulations

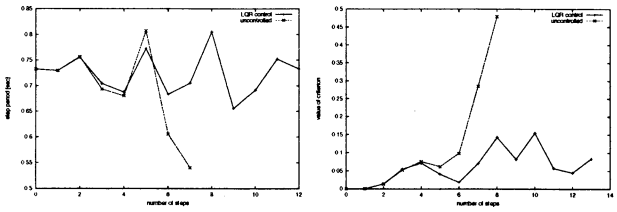
We consider a criterion function as the value of difference between the actual value and its nominal value, that is:

$$J_k := \|\boldsymbol{q}_k - \boldsymbol{q}^*\|_2. \tag{9}$$



(a) step period (b) norm

Figure 2: Stabilizing control on nominal slope



(a) step period (b) norm

Figure 3: Stabilizing control on rugged slope

So the value of criterion is smaller means the performance of the control law is better.

Fig.2 and Fig.3 are simulation results of stabilizing control.

Fig.2 is the result of stabilizing control on nominal slope, that is, without disturbances. The robot start from following initial condition:

$$\boldsymbol{q}_0 = \left[\begin{array}{cccc} -0.250 & 0.350 & 1.20 & 1.20 \end{array} \right]^T$$

which converges stable limit cycle and make the robot walk within 10[sec]. From (a), it is clear that step period converges actual value quickly with control of gravity field. From (b), the value of criterion also converges to 0 quickly in the case of controlled.

Fig.3 is the result of stabilizing control on rugged slope where $\sigma = 0.0050$. The robot start from actual condition:

$$\boldsymbol{q}^* = \left[\begin{array}{cccc} -0.21661 & 0.31661 & 1.08355 & 0.39932 \end{array} \right]^T$$

and make the robot walk within 10[sec]. From (a), in the case without control, since the robot cannot continue passive walking, the data is stopped. In the case with control, it can be seen that the robot can continue passive walking. From (b), the value of criterion is without control, but the value is kept in bounded area with control. We can see the validity of control law.

The final mark of this research is to stabilize active walking robot walking on the level. In the following, the application method is explained.

4 Application to Active Walking

4.1 Virtual passive walking on the level

Physical model is shown in Fig.4. The model is compass-like biped robot which has actuators at ankle and hip. (Fig.4 right) Let us consider virtual gravity field $g \tan\gamma$ in the horizontal direction as shown in Fig.4 left, then the robot exhibits passive walking virtually without control forces and the gait pattern of the walking robot converges steady limit cycle. In other words, the condition can be considered as normal gravity field of $\frac{g}{\cos\gamma}$ and the robot walk down shallow slope γ.

We use the walking pattern as desired model of active walking robot which walks on the level by actuators. It is possible to tranceform the virtual gravity effect to torque of hip and ankle actuators. In this paper, we set $\gamma = 0.02$ [rad]. Since the value of γ is very small, the condition can be considered to be very closed to real condition, so the gait pattern can be considered to be *natural*.

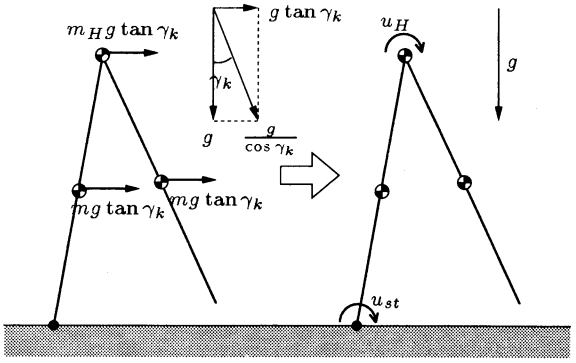


Figure 4: Virtual gravity field and torque tranceformation

4.2 Torque tranceformation

The dynamic equation of the real robot on the level:

$$\boldsymbol{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + \boldsymbol{g}(\boldsymbol{\theta}) = \boldsymbol{\tau} + \boldsymbol{F}_e. \quad (10)$$

Where $\boldsymbol{\tau} = \boldsymbol{S}\boldsymbol{u}$ is control input and $\boldsymbol{F}_e$ is the environmental forces. Then the control torque is given by:

$$\boldsymbol{\tau} = \boldsymbol{S}\boldsymbol{u} = \left[\begin{array}{cc} 1 & 1 \\ 0 & -1 \end{array} \right] \left[\begin{array}{c} u_{st} \\ u_H \end{array} \right] \quad (11)$$

$$= \begin{bmatrix} (m_H l + ml + ma)\cos\theta_{st} \\ -mb\cos\theta_{sw} \end{bmatrix} g\tan\gamma_k \quad (12)$$

and torque of ankle and hip are given by:

$$u_{st} = (m_H l + ml + ma)g\cos\theta_{st}\tan\gamma_k - mbg\cos\theta_{sw}\tan\gamma_k \quad (13)$$

$$u_H = mbg\cos\theta_{sw}\tan\gamma_k \quad (14)$$

Since $\boldsymbol{u} = \boldsymbol{u}(\boldsymbol{\theta}, \gamma_k)$ consists of continuous time information $\boldsymbol{\theta}$ and discrete time information γ_k, the control can be considered as hybrid control.

4.3 Active walking on the level

The above control can stabilize the walking robot on the rugged level ground and realize small-torque control.

If the walking robot was pushed toward the walking direction, the walking motion will change quickly because energy of the robot system changes. We examined the effect of such environmental forces.

Here, we consider "virtual energy" which is total energy of the robot defined in virtual gravity field. Virtual energy E_{VIR} is sum of kinetic energy and potential energy in virtual gravity field, that is,

$$E_{\mathrm{VIR}} = K_{\mathrm{VIR}} + P_{\mathrm{VIR}} \quad (15)$$

K_{VIR} and P_{VIR} are defined as follows:

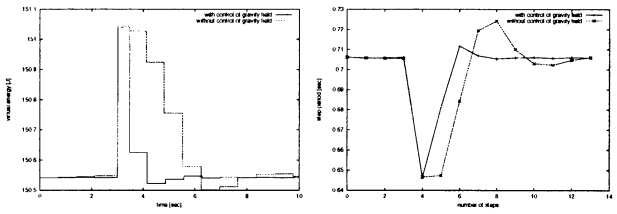
$$K_{\mathrm{VIR}} = K(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \frac{1}{2}\dot{\boldsymbol{\theta}}^T \boldsymbol{M}(\boldsymbol{\theta})\dot{\boldsymbol{\theta}} \quad (16)$$

$$= \frac{1}{2}(m_H l^2 + ma^2 + ml^2)\dot{\theta}_{st}^2 + \frac{1}{2}mb^2\dot{\theta}_{sw}^2 - mbl\cos(\theta_{st} - \theta_{sw})\dot{\theta}_{st}\dot{\theta}_{sw} \quad (17)$$

$$P_{\mathrm{VIR}} = (m_H l + ml + ma)\hat{g}\cos(\theta_{st} + \gamma_k) - mb\hat{g}\cos(\theta_{sw} + \gamma_k) \quad (18)$$

Where $\hat{g} = \frac{g}{\cos\gamma_k}$. Please notice that it implies "virtual passivity". In other words, if $\boldsymbol{F}_e = \boldsymbol{0}$, E_{VIR} is kept constant.

Fig.5 shows the simulation results. The robot is pushed toward the horizontal direction with 100.0 [N] during $3.0 \leq t \leq 3.01$. At $t = 3.0$ [sec], from (a), in each case virtual energy of the model increases rapidly and maintain constant value again from the instant of $\boldsymbol{F}_e = \boldsymbol{0}$. In the case of virtual gravity field control, we can see that virtual energy is controlled to actual value. From (b), step period changes after pushing, that is, the walking speed changes. In the case with control of virtual gravity field, step period is also controlled to actual value.



(a) virtual energy (b) step period

Figure 5: Effect of environmental forces

5 Summary

In this paper, we proposed a gait design method using virtual passive walking with stabilizing control of gravity field. In other words, active walking on the level is realized without any gait design. The validity of the control law has investigated by computer simulation. The robot can walk on rugged slope or the level with only control of virtual gravity field. The walking system is constructed without injuring "virtual passivity".

In the future work, we should apply the control law to experimental machine.

Acknowledgment

This work was partially supported by the Scientific and Research Foundation of the Ministry of Education under Grant COE #104723.

References

[1] Goswami A, Espiau B, Keramane A (1996), Compass-like biped robot Part I: Stability and bifurcation of passive gaits, Research report INRIA, No.2613, 1996.

[2] Goswami A, Espiau B, Keramane A (1997), Limit cycles in a passive compass gait biped and passivity-mimicking control laws, Autonomous Robots 4, pp.273-286, 1997.

[3] McGeer T (1990), Passive dynamic walking, The Int. J. of Robotics Research, 9, pp.62-82, 1990.

[4] Asano F, Yamakita F, Furuta K (1999), A gait design of humanoid robot using passive biped walking, Proc. of the 38th SICE Annual Conference Domestic Session Papers Vol.II, pp.413-414, 1999.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Control of a Snake Robot with Redundancy Based on Kinematic Model

Kazutaka Mogi
Dept. of Computational Intelligence
and Systems Science
Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku,
Yokohama 226-8502, JAPAN

Fumitoshi Matsuno
Dept. of Computational Intelligence
and Systems Science
Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku,
Yokohama 226-8502, JAPAN

Abstract

In this paper we define the redundancy controllable system and we consider control of a redundant snake robot based on the wheeled link model. We derive a condition and a control law for the snake robot that the system becomes redundancy controllable. From simulation results we find that the crowing motion of the snake robot is natural.

1 Introduction

Unique and interesting gait of the snakes makes them able to crawl, climb a hill, climb a tree by winding and move on very slippey floor. It is useful to consider and understand the mechanism of the gait of the snakes for mechanical design and control law of snakes robot.

Hirose has long investigated snake robots and produced several snake robots, where he modeled the snake by a link mechanism with no side slip[1]. The feedback control law for the snake head's position using Lyapunov method has been developed by Mita on the basis of the dynamic model[2]. He pointed out the controller can stabilize the head position of the snake robot to its desired value, but the configuration of it converges to a singular configuration. From the model we find that the snake robot dose not have the redundant degrees of freedom, and this leads to the difficulty in the control objective of the singular configuration avoidance.

In this paper we define the redundancy controllable system. We find that introduction of links without wheels and shape controlled points in the snake robot's body makes the system redundancy controllable. In this case the head's velocity of the snake robot does not determine all joint velocities of the robot uniquely. On the basis of the control technique for the redundant manipulator [3] we introduce the cost function correspond to the measure for the singularity and manipulability of the system, and construct a controller with considering the redundancy. Using redundancy, it becomes possible to accomplish both main objective of controlling position and posture of the snake robot head, and also sub-objective of the singular configuration avoidance and the obstacle avoidance. Simulations for a 6-link snake robot with three wheeled link are shown.

2 Redundancy Controllable System

Let $\boldsymbol{q} \in R^n$ be generalized coordinates, $\boldsymbol{u} \in R^p$ be the input, $\boldsymbol{w} \equiv S\boldsymbol{q} \in R^q$ be the state vector to be controlled, S be a selection matrix, whose row vectors are independent unit vectors, related to generalized coordinates. We define that the system

$$A(\boldsymbol{q})\dot{\boldsymbol{w}} = B(\boldsymbol{q})\boldsymbol{u} \tag{1}$$

is redundancy controllable if $p > q$, the matrix A is full column rank, B is full row rank, and there exists an input $\boldsymbol{u}$ which accomplishes both main objective of the convergence of the vector $\boldsymbol{w}$ to the desired state $\boldsymbol{w}_d$ $(\boldsymbol{w} \to \boldsymbol{w}_d, \dot{\boldsymbol{w}} \to \dot{\boldsymbol{w}}_d)$ and sub-objective of increase (or decrease) of the cost function $V(\boldsymbol{q})$.

For a snake robot based on the wheeled link model we discuss a condition that the system is redundancy controllable.

3 Kinematic Model of Snake Robot

We consider a redundant n-link snake robot as shown in Fig.1. Let n be the number of links, m be the number of wheeled links, $[\,x_h \quad y_h \quad \theta_h\,]^T$ be the vector of the snake head and posture, $[\,\phi_1 \quad \cdots \quad \phi_{n-1}\,]^T$

be the vector of relative joint angles and $\boldsymbol{q} = [x_h \quad y_h \quad \theta_h \quad \phi_1 \quad \cdots \quad \phi_{n-1}]^T$ be generalized coordinates.

The wheels are located on the middle point of each link whose length is $2l$. Let $[x_i \quad y_i]^T$ be the position vector of the middle point of the link i. As the wheel does not slip to side direction, the velocity constraint condition should be satisfied. If the i-th link is wheeled, the constraint can be written as

$$\dot{x}_i \sin(\theta_h + \sum_{h=1}^{i-1} \phi_h) - \dot{y}_i \cos(\theta_h + \sum_{h=1}^{i-1} \phi_h) = 0. \quad (2)$$

From the geometric relation the position vector is expressed as

$$\begin{aligned} x_i &= x_h + 2l\cos\theta_h + 2l\sum_{k=1}^{i-2}\cos(\theta_h + \sum_{j=k}^{i-2}\phi_j) \\ &\quad + l\cos(\theta_h + \sum_{j=k}^{i-1}\phi_j) \quad (3) \\ y_i &= y_h + 2l\sin\theta_h + 2l\sum_{k=1}^{i-2}\sin(\theta_h + \sum_{j=k}^{i-2}\phi_j) \\ &\quad + l\sin(\theta_h + \sum_{j=k}^{i-1}\phi_j). \quad (4) \end{aligned}$$

Substituting (3), (4) into (2), gives the velocity constraint equation.

As one wheeled link has one velocity condition, the number m of the wheeled link is equal to the number of equations. To satisfy the condition (B is full row rank) that the system is redundancy controllable, we assume the wheeled link has an active joint where an actuator is mounted. Other joints are passive or active. The passive joint angles are defined as a state variable to be controlled as shape controlled points. Let s be the number of passive joits. It is equivalent to the number of shape controlled points in the snake robot body. We call the number s the shape controllable index. The angular velocity $\dot{\boldsymbol{\theta}}$ of the active joints is regarded as the control input $\boldsymbol{u}$. As not only the position of the snake robot head but also the posture are controlled, the first link should be wheel free. If not, there exists possibility that the matrix B is not full row rank for a special configuration. We find that the movement of links behind the last wheeled link does not contribute the movement of the snake robot head. We assume that the tail link is a wheeled link.

The state vector $\boldsymbol{w}$ to be controlled has the elements of x_h, y_h, θ_h and angles of all passive joints. We find that exclusion of $\boldsymbol{w}$ from $\boldsymbol{q}$ gives $\boldsymbol{\theta}$. $\{\boldsymbol{w}\} \cup \{\boldsymbol{\theta}\} = \{\boldsymbol{q}\}, \{\boldsymbol{w}\} \cap \{\boldsymbol{\theta}\} = \boldsymbol{\phi}$. We can define $\boldsymbol{w} = S\boldsymbol{q}$, $\boldsymbol{\theta} = \bar{S}\boldsymbol{q}$. From (2)-(4) we obtain the state equation

$$A(\boldsymbol{q})\dot{\boldsymbol{w}} = B(\boldsymbol{q})\boldsymbol{u}, \quad \boldsymbol{u} = \dot{\boldsymbol{\theta}} \quad (5)$$

where $A \in R^{m\times q}, B \in R^{m\times p}$. We design the system so as to satisfy $m \geq q$, if not the input $\boldsymbol{u}$ does not determine the velocity $\dot{\boldsymbol{w}}$ of the state vector to be controlled uniquly. We set $m < p$ so as to satisfy the condition that the velocity vector $\dot{\boldsymbol{w}}$ does not determine the control input $\boldsymbol{u}$ uniquely (i.e. the input is redundant).

From two conditions $m \geq q$ and $m < p$ the condition $p > q$ is satisfied. The two conditions can be written as

$$3 + s \leq m < (n-1) - s. \quad (6)$$

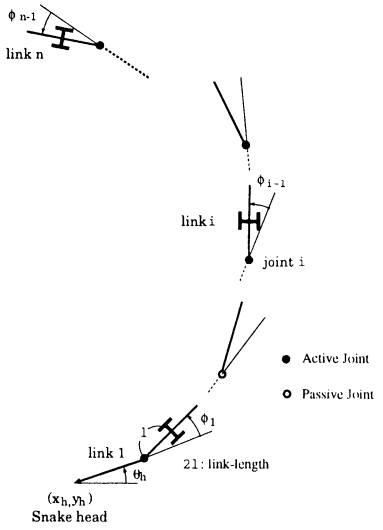


Fig.1 A redundant n-link snake robot

4 Controller Design

Let us define the control input as follows:

$$\boldsymbol{u} = B^*A\{\dot{\boldsymbol{w}}_d - K(\boldsymbol{w} - \boldsymbol{w}_d)\} + (I_k - B^*B)\alpha\boldsymbol{\eta} \quad (7)$$

where B^* is a pseudo-inverse matrix of B, $\boldsymbol{\eta} = \nabla_{\boldsymbol{\theta}}V(\boldsymbol{\theta}) = [\partial V/\partial\theta_1 \cdots \partial V/\partial\theta_{n-1-s}]$ is the gradient of the cost function $V(\boldsymbol{\theta})$, and $\alpha \geq 0$, $K > 0$. The first term of the right side of (7) is the control input term to accomplish main objective of the convergence of the state vector $\boldsymbol{w}$ to be controlled. As the second term $(I - B^*B)\alpha\boldsymbol{\eta}$ belongs to the null space of the matrix B, it does not disturb the dynamics of the controlled vector $\boldsymbol{w}$. As there is no interaction between $\boldsymbol{w}$ and $\boldsymbol{\theta}$, we

find that the control accomplishes the sub-objective. In this simulation we set $B^* = B^T(BB^T)^{-1}$ and

$$V = a'(det(A^T A)) + b'(det(BB^T)) \tag{8}$$

where $a', b' > 0$. The first term of the right side of (8) implies the measure of the singular configuration. A set of joint angles which satisfies $rankA < q$ (A is not full column rank) means the singular configuration, for example a straight line ($\phi_i = 0, i = 1, \cdots, n-1$). The second term of the right side of (8) is related to the manipulability of the system.

The closed-loop system is expressed as

$$A\{(\dot{\boldsymbol{w}} - \dot{\boldsymbol{w}}_d) + K(\boldsymbol{w} - \boldsymbol{w}_d)\} = 0. \tag{9}$$

As the matrix A is full column rank, the controller ensures the convergence of the controlled state vector to the desired value ($\boldsymbol{w} \longrightarrow \boldsymbol{w}_d$).

5 Simulation

5.1 Movement of the Snake Robot

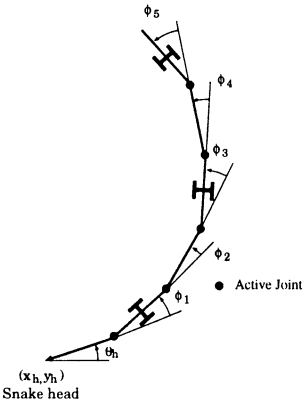


Fig.2 A 6-link snake robot

For simplicity we consider a 6-link snake robot with 3 wheeled links and does not introduce the shape controlled points ($s = 0$) (Fig.2). In this case the matrix A is square. We set the initial condition $w(0), \theta(0)$ and the desired condition $w_d(t)$ as $w(0) = [0 \quad 0 \quad \frac{3\pi}{4}]^T$, $\theta(0) = [\frac{\pi}{100} \quad \frac{\pi}{90} \quad \frac{\pi}{80} \quad \frac{\pi}{70} \quad \frac{\pi}{60}]^T$, $w_d = [t \quad 0 \quad \pi]^T$, $\dot{w}_d = [1 \quad 0 \quad 0]^T$, and $l = 0.05[m]$, $K = diag(3,3,3)$. We set coefficients of the cost function V as $a' = a/l^2, b' = b/l^6$ in order to normalize with respect to the link length l. Figs. 3-5 show the transient responses. The left column in each figure shows transient responses for x_h, y_h, θ_h, $detA/l$, $\sqrt{det(BB^T)}/l^3$ and the right column shows transient responses for $u_1, \cdots, u_5$.

Fig.3 shows the responses for $\alpha = 0$ (*case* 1). In this case the controller does not use redundancy. From the figure we find that the snake head tracks the desired state, but $detA$ converges to zero. In *case* 2 and 3 we set $\alpha = 9$. In these cases the controller uses redundancy so as to attain the increase of V. We set $a = 5, b = 1$ (*case* 2) and $a = 2, b = 1$ (*case* 3). In Figs.4 and 5 the snake head converges to the desired state. From Fig.4 (*case* 2) the avoidance of the singular point is accomplished, but the amplitude of the input is very big and the frequency of the input is too high. It is impossible to realize the obtained input. From Fig.5 (*case* 3) the snake robot becomes singular about 15 times per second, but the amplitude and frequency of the input are acceptable. Fig.6 shows the movement of the snake robot for each case. We find the crawling of the snake in the *case* 3. In the *case* 1 the snake robot converges to a singular configuration of the straight line. In the *case* 2 the movement of the snake robot is quite far from that of natural snakes.

We find that the crawling movement of the snake robot around the singular configuration of the straight line is natural in view of the restriction of the input power and that redundancy is important for the movement.

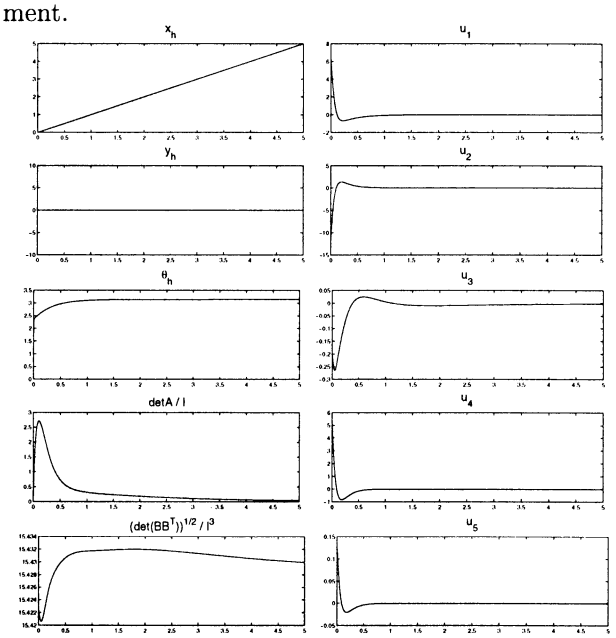
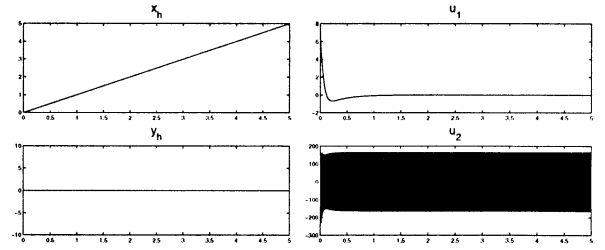


Fig.3 Transient responses for controller without considering redundancy ($\alpha = 0$)(*case* 1)



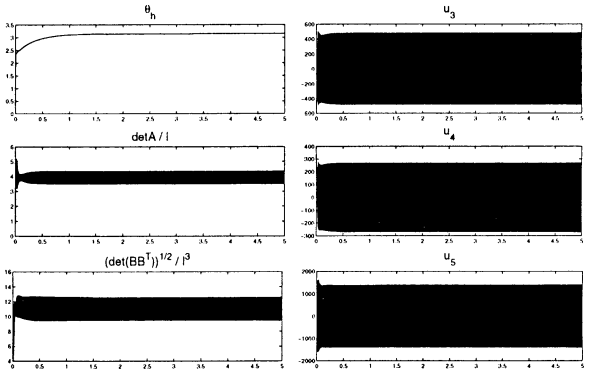


Fig.4 Transient responses with considering redundancy and big weight for singularity avoidance ($\alpha = 9, a = 5, b = 1$)($case$ 2)

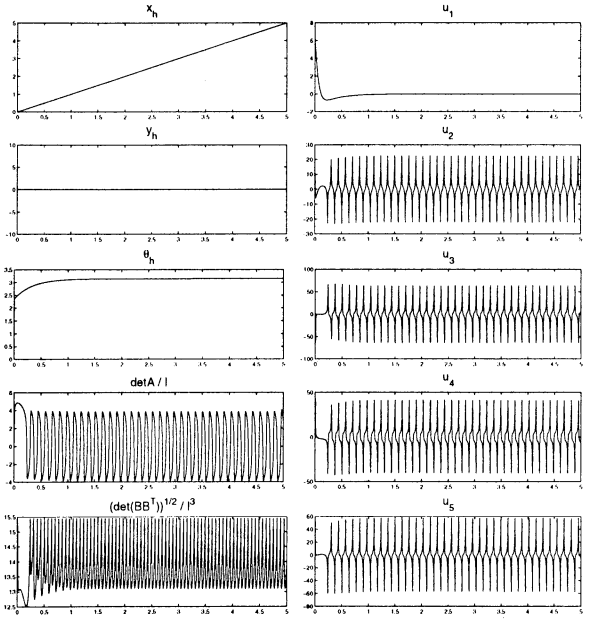


Fig.5 Transient responses with considering redundancy and small weight for singularity avoidance ($\alpha = 9, a = 2, b = 1$)($case$ 3)

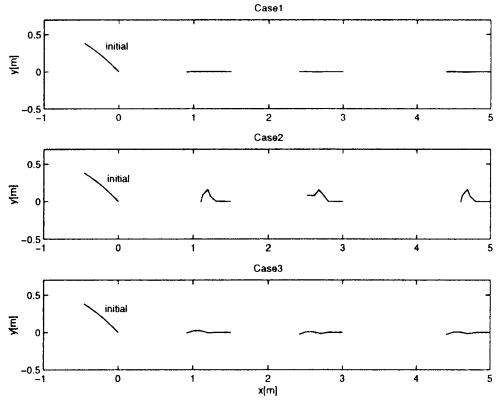


Fig.6 Movement of the snake robot

5.2 Obstacle Avoidance

We set the desired condition $w_d(t)$ so as to avoid the obstacles. Fig.7 shows the movement of the snake robot for the obstacle avoidance. In $case$ 1 we set $\alpha = 0$, and the controller does not use redundancy. The snake robot converges to a singular configuration of the straight line, so can not track the desired trajectory and hits the obstacles. In $case$ 2 we set $\alpha = 9, a' = 2, b' = 1$. We find that the snake robot can realize the crawling motion and track the desired trajectory to avoid the obstacles.

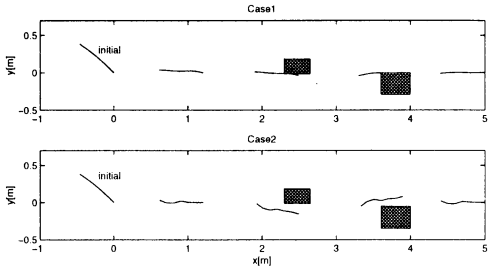


Fig.7 Obstacle avoidance

6 Conclusion

We consider the condition that the system for the snake robot becomes redundancy controllable, and we introduce wheel free links in the snake robot body. Using redundancy, it becomes possible to accomplish both main objective of controlling the position and the posture of the robot head and the shape controlled points, and also sub-objective of the singular configuration avoidance and the obstacle avoidance. We find that the crawling movement of the snake around the singular configuration of the straight line is natural in view of the restriction of the input power.

As the future works, we should expand the obtained results to the dynamic model.

References

[1] S.Hirose, Biologically Inspired Robots (Snake-like Locomotor and Manipulator), Oxford University Press, 1993

[2] P.Prautesch, T.Mita, H.Yamauchi, T.Iwasaki and G.Nishida, Control and Analysis of the Gait of Snake Robots, Proc. COE Super Mechano-Systems Workshop'99, 257-265, 1999

[3] T.Yoshikawa, Analysis and Control of Robot Manipulators with Redundancy, Robotics Research The First International Symposium, MIT Press, Cambridge, Mass. 735-747, 1984

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Autonomous Formation of a Mechanical Structure by a Cellular Automaton

Norio Inou, Kyosuke Okumura and Sadayuki Ujihashi

Tokyo Institute of Technology, Department of Mechanical and Environmental Informatics,
O-okayama, Meguro-ku, Tokyo 152-8552, Japan
E-mail: inou@mei.titech.ac.jp

Abstract: This paper describes a cellular automaton model including mechanical properties based on remodeling of living tissues. The proposed model consists of identical elements which mechanically connect each other. Each element senses mechanical states and changes Young's modulus itself. The performance of the model is examined by computer simulations of three-dimensional cantilever problems. The computer simulations show the following results. The model has an ability to generate various biomimetic topological structures which are almost satisfied with fully stress design. The mechanical formation is uniquely determined once the mechanical conditions are given. The initial arrangement of cells, initial distribution of Young's modulus and aspect ratio of the active space of cells greatly affect the topological formation. It is difficult to predict the final topological structure although the initial mechanical conditions shall be determinately given.

Key words: Cellular Automaton, Remodeling, Self-organization, Biomimetic Structures

1 INTRODUCTION

Living tissues such as bone or wood have remarkable features to flexibly adapt to circumstances. The adaptive function called remodeling or functional adaptation has an important role to self-organizing a structure of tissues. The remodeling process is a kind of local feedback mechanism. In case of bone, when a stress at a part of bone is not set up for a long time, the part is gradually weakened. Conversely, when a part of bone is highly stressed, the part is strengthened. As the result of continuous process, the whole bone forms a proper structure to the mechanical environment[1].

From a viewpoint of engineering, the adaptive formation process of remodeling can be explained by an autonomous decentralized model. It consists of identical autonomous elements which sense mechanical states and change the material properties themselves as shown in Fig. 1. This is a sort of a cellular automaton which consists of autonomous cells. Cellular automata are generally known as pattern generators. The proposed model has a new feature to generate mechanical structures as each cell includes mechanical properties.

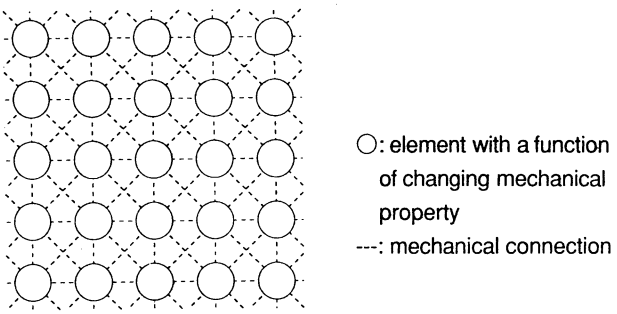


Fig. 1 A cellular automaton including mechanical properties

2 SIMULATION METHOD

Performance of the proposed model was examined by the following iterative computation.

Step 1. Give an acting space of cells.
Step 2. Set loads and boundary conditions on cells.
Step 3. Calculate stresses produced in the cells with the finite element method.
Step 4. Change Young's modulus of the cells.
Step 5. Apply "death" and "birth" rules to the cells.
Step 6. Return to Step 2.

In the step 1, the acting space of cells is a design area of mechanical structure as shown in Fig. 2. A square of element indicated by a dark tone corresponds to a cell of the automaton. These cells can act within the area indicated by a light tone, but can not act in other areas. In the step 2 and 3, a stress analysis for each iteration is performed by the finite element method to obtain stresses produced in the cells. And in the step 4, Young's modulus E of each cell is independently changed by the following equation.

$$E^{(t+1)} = E^{(t)} \{ 1+ \alpha (\sigma - \sigma_c) / \sigma_c \} \quad (1)$$

Where, t: iteration steps, σ: equivalent stress, σ_c: target stress, α: constant value about 0.1.

Iterative calculations will lead the converged state that σ is nearly equal to σ_c for almost cells. As the target stress corresponds to the cell's strength, the converged structure is expected to be mechanically rational. Here, we must notice the relationship between the target stress and Young's modulus of the cell because the target stress changes its value with that of Young's modulus. Expecting that Young's moduli of cells are converged to a value and the final structure is satisfied with fully stress design, we set a non-linear relationship between the target stress and the Young's modulus as shown in Fig. 3.

In the step 5, after changing Young's moudli of cells, "death" and "birth" rules are applied to the cells for the purpose of generating a clearer fine structure. The rules give the following characters like "life game" proposed by Conway[2]. When Young's modulus of a cell decreases below a value, the cell dies (or vanishes). Conversely, if a cell is highly stressed, a new cell is born at a vacant place near the cell. Mattek proposed a structural optimization method based on remodeling of wood[3]. His method has SKO (soft kill option) which corresponds to the "death" rule, but does not include the "birth" rule.

In our simulation method, the structural formation is proceeded by partial modification. As far as the partial modification of structural shape, similar methods have been proposed in the field of structural optimization[4]. Almost these methods are based on sensitive analysis with global evaluation. Our simulation method, however, does not include any global estimation because the method is based on an autonomous decentralized system.

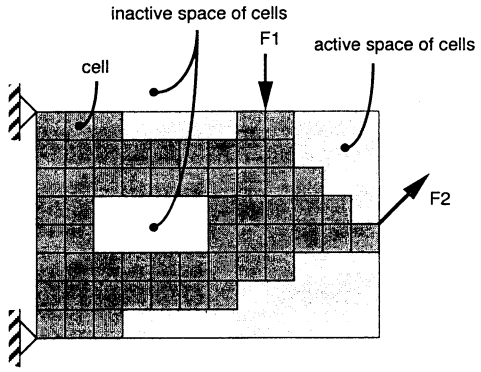


Fig. 2 Layout of cells and mechanical conditions.

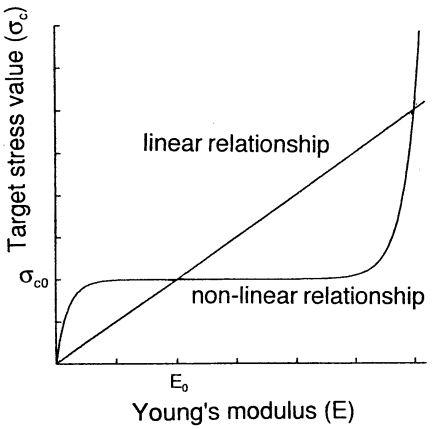


Fig. 3 Non-linear relationship between a target stress (the cell's strength) and the Young's modulus.

3 SIMULATION RESULTS

In our previous reports[5,6], two-dimensional simulations were discussed. In this research, three-dimensional simulations were examined from the following

viewpoints: formation process of a structure, effect of the initial arrangement of cells and effect of aspect ratio.

3.1 Formation process of a structure

To examine the performance of the proposed model, a cantilever problem was set as shown in Fig. 4. The initial active space is a rectangular prism composed of 40x20x20 cells (scale: 200x100x100mm) and the initial Young's modulus E_0 was unified to be 100 GPa. The initial target stress σ_{c0} for E_0 was set as 30 MPa. The loading force was set as 2.0×10^4 N. Fig. 5 shows the simulation results. From a rectangular shape with the homogeneous distribution of Young's modulus, a three-dimensional topological structure was generated.

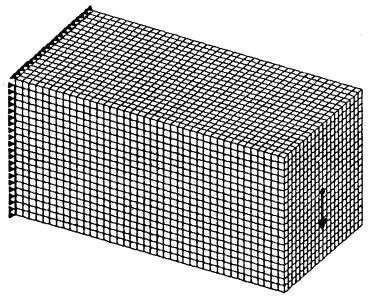


Fig. 4 A cantilever problem

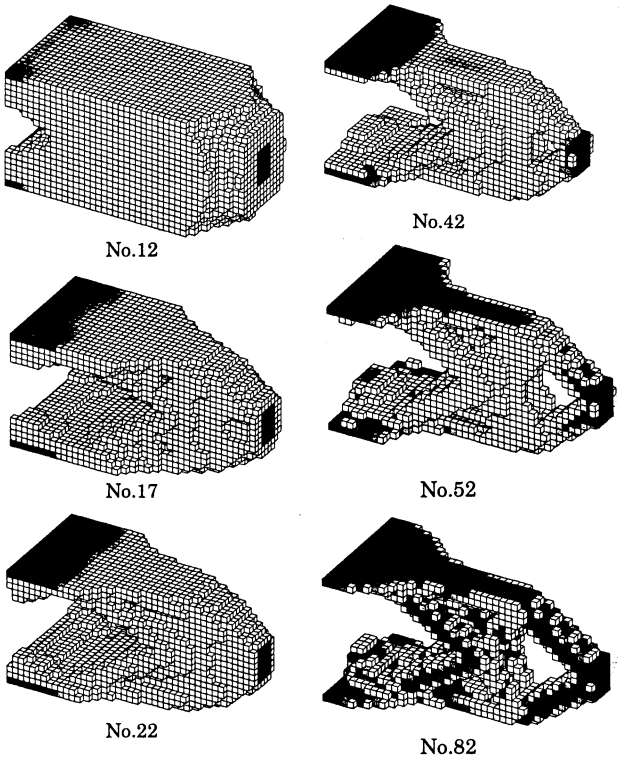


Fig. 5 Formation process of the structure.
Cells with a dark tone are almost in convergent states.
Numbers in the figures denote iteration steps.

3.2 Effect of initial arrangement of cells

The proposed model self-organizes an adaptive structure from any initial arrangement of cells in the active space. From the two-dimensional simulations in the previous studies[5,6], it is expected that the initial arrangement of cells or the initial distribution of Young's modulus affects the structural formation.

Several different arrangement of cells and the distributions of Young's modulus were provided. Fig. 6 shows some of the results. These final structures were uniquely determined once the initial conditions were given. However, it is very difficult to predict the final structure without executing the computer simulation.

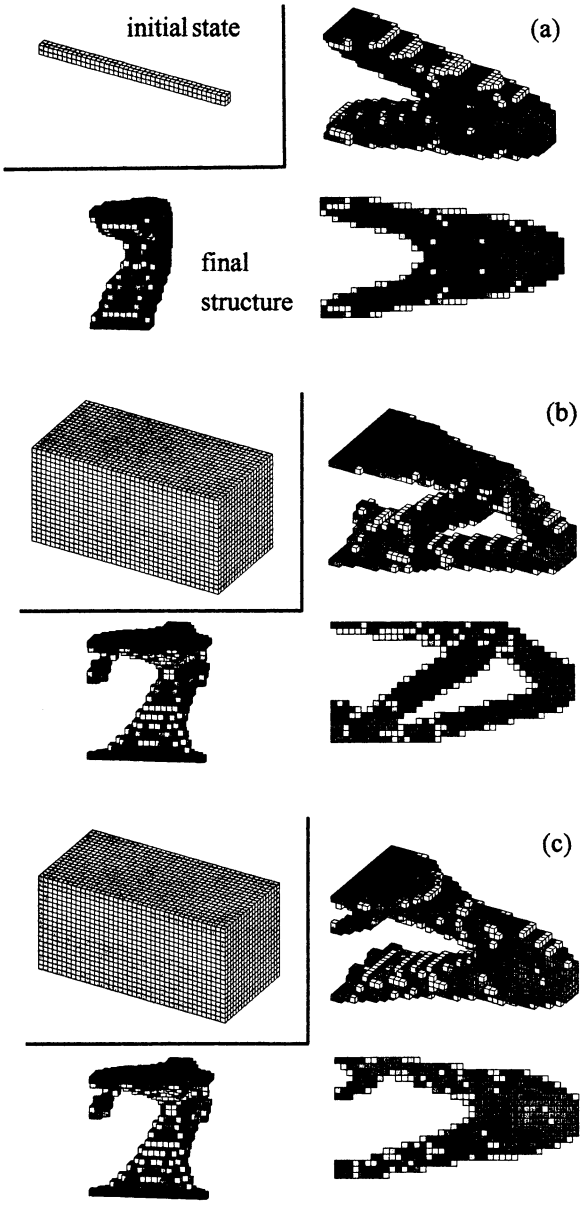
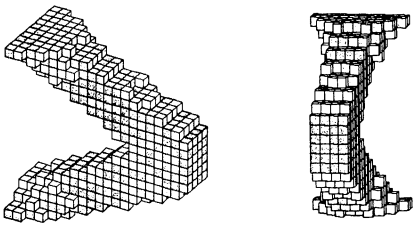


Fig. 6 Initial states of cells and the final structures.

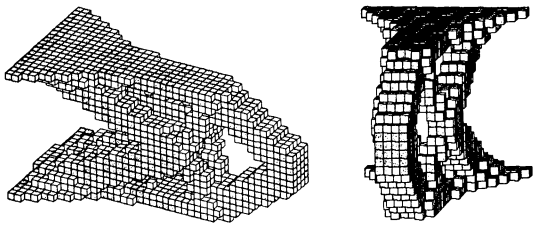
3.3 Effect of aspect ratio of active space

Aspect ratio of the active space is also important factor in the structural formation. Cantilever problems were also applied in this subject. The results are shown in Fig. 7. When the ratio was 1.0, a simple straight beam structure with only two-dimensional extent was formed. However, when the ratio was 2.0 or 3.0, the final structure became a complicated topological structure with three-dimensional extent. The structures are similar to trabeculae of a bone[7].

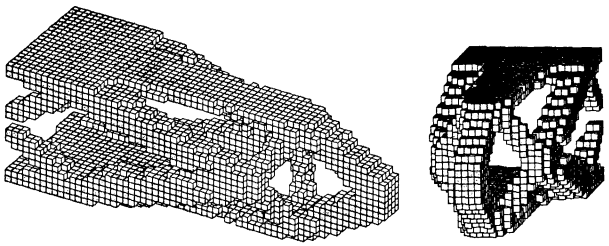
It is considered that the three-dimensional extent is caused by effect of produced stresses around the fixed plane. When the aspect ratio is small, the medial position at the top (or bottom) of the fixed plane clearly shows a large stress. In this case, cells around the medial position predominantly increases the Young's modulus, which will lead to two-dimensional structural formation. With increase of the aspect ratio, the stress distribution becomes uniform around the fixed plane because difference of moment produced by the loading force decreases. Therefore, the medial cells lose the predominance, which will lead to a three-dimensional extent.



aspect ratio = 1.0



aspect ratio = 2.0



aspect ratio = 3.0

Fig. 7 Effect of aspect ratio of active space.

4 CONCLUSIONS

A cellular automaton model including mechanical properties was proposed. Characteristics of the model in three-dimensional cases were examined. The initial arrangement of cells affects the structural formation as well as two-dimensional cases. The aspect ratio of initial shape is also an important factor because the development of structural formation is affected by the active space of cells.

Acknowledgement: The authors appreciate the help of Mr. Todoroki K, who assisted in performing computer simulation. This study was supported by a Japanese Grant-in-Aid for COE Research Project supported by the Ministry of Education, Science, Sports and Culture, "Super Mechano-Systems" (No. 09CE2004).

REFERENCES

[1] Kummer B (1972), Biomechanics of Bone, in Fung YC, Perrone N, and Anliker M (eds.), Biomechanics, Englewood Cliffs, New Jersey, pp.237-271.

[2] Gardner M (1970), Mathematical Games, Scientific American, Vol.223, pp.120-123.

[3] Mattek C (1997), Design in Nature, Springer.

[4] Bendsøe MP and Kikuchi N (1988), Generating Optimal Topologies in Structural Design Using a Homogenization Method, Comput. Meths. Appl. Mechs. Engrg., 71, pp.197-224.

[5] Inou N, Shimotai N and Uesugi T (1994), A Cellular Automaton Generating Topological Structures, Second European Conference on Smart Structures and Materials (Glasgow/U.K.), pp.47-50.

[6] Inou N, Uesugi T, Todoroki K and Ujihashi S (1995), Autonomous Formation of Biomimetic Structures by Cellular Automaton, Proc. 4th Japan International SAMPE Symposium, pp.621-626.

[7] Alexander RM (1994), Bones, The Unity of Form and Function, Weidenfeld&Nicolson.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

An Evolutionary Optimization Method for Structural Dynamics
- To optimize the structural components of Super Mechano-Systems -

Masaaki Okuma
Associate professor, Department of Mechanical Engineering and Science, Tokyo Institute of Technology.
Hiroki Sakuma
Power System Headquarters, Mitsubishi Heavy Industry, Ltd.
Kenji Hashidume
Graduate student, Department of Mechanical Engineering and Science, Tokyo Institute of Technology.

Abstract:
In this paper, the authors present a new evolutionary optimization method for structural dynamics and statics. This method finds the optimum shapes of structural components and systems for dynamic and/or static loads. The method not only optimizes the size parameters of structures but also optimizes the topology of the shapes. The optimized structural design can be obtained from a simple and small initial structure. The initial shape of a structure is modeled by identical finite elements. The surface of the model is covered with a layer of the virtual finite elements. Here, the virtual finite elements are those which are formulated with negligibly small material properties. The deformation of the model by applying external loads is computed with the finite element analysis. Then, according to an algorithm developed based on modal analysis, some of the virtual elements will change to real elements, and some real elements will be deleted. This computational process is iterated. Eventually, the structure is optimized. In order to demonstrate the usefulness of the method, the optimization of a simple 2-D structure is presented.

Keywords: optimization method; structural dynamics; finite element analysis; 2-D structure

1. Introduction

In this paper, the authors present a new evolutionary optimization method for structural dynamics and statics. This method finds the optimum shapes of structural components and systems for dynamic and/or static loads. The method is not only optimizes the size parameters of structures but also optimizes the topology of the shapes. The optimum structural design for dynamics can be obtained from a simple and small initial structure by an algorithm based on modal analysis. Therefore, comparatively less computational cost and time will be required to obtain results than conventional ones.

The last decade, some methods categorized as evolutionary optimization methods have been developed. The homogenization method[1] is a famous evolutionary optimization method. The homogenization method fills the whole feasible design domain with many micro-cells. The size parameters of the holes in the micro-cells are adopted as design variables. Then, the computation becomes costly. The cellular automaton method[2] will be another method. This method may originate in adaptive shape optimization methods[3] for structural material using the growing-reforming procedure, such as the surface growth method and the thickness variation method. The cellular automaton method optimizes the shape of a structure by the computation, regarding only local reaction process. The model has an ability to generate various biomimetic topological structures which are almost satisfied with fully stress design. Therefore, this method works usefully for static analysis, but not for dynamic analysis such as the optimization of controlling natural frequencies. In addition, the demonstrations presented in most of those papers were started from a special initial condition. That is, the whole feasible design domain is modeled with lots of cells as the initial structure. It will be more practical if an optimum design can be obtained from simple and small initial structures. Because engineers can create good mechanical structures as initial ones.

In order to create very sophisticated robotic machines in the 21st century, both control systems and structural systems should be optimized much more with robustness. This is clear with an example talk about sports players. Excellent control skill of the body action/motion and an optimized body are indispensable for a very good/strong sports player. The optimized body here means the appropriate constitution with the muscles that can demonstrate the necessary power and with no useless superfluous flesh. Sport players do train their control skill and obtain their optimized bodies by good practice. The improvement of their bodies can be considered as a sort of the evolutionary optimization of

human bodies.

The structural optimization method presented in this paper may be categorized as a surface growth method. It will work usefully for the determination of good structural components and systems of robotic machines. Engineers can design the initial structural components. Therefore, it is important to develop the computational methods for optimizing those initial designs. The method presented in this paper is suitable from this viewpoint.

2. Algorithm

The outline of the algorithm is as follows. The initial shape of a structural component or a system is modeled using identical finite elements. The surface of the model is covered with a layer of virtual finite elements. Virtual finite elements are formulated using negligibly small material constants, which is, a much smaller material density and an elastic modulus than the actual material's values of the structure. The deformation of the structure at a resonant frequency is computed with the finite element analysis. For each finite element, including the virtual elements, the following physical quantities are calculated: Strain energy and kinetic energy stored in each element under the deformation. According to the theory of the modal analysis, a resonant frequency of the structure can be determined based on the ratio of strain energy and kinetic energy stored in the whole of the structure with respect to its associating resonant vibration mode. Then, in the case of optimum design of controlling the resonant frequencies, it can be determined which virtual finite element should be changed to a real finite element and which real finite element should be deleted. This computational process is iterated, subject to the constraint of the maximum number of finite elements available to use. Eventually the optimum shape of the structure can be obtained. The optimization will be considered as a minimum weight optimization of the initial structure. In this research work, the yield criterion of an individual element is judged by THE Von Mises yield criterion in the cases of static analysis and forced vibration analysis.

The algorithm of the method is explained with the flowchart shown in Fig.1. At first, an initial structure is made of finite elements. All finite elements are identical. Next, the structure is covered with a single layer of virtual finite elements. Here the virtual finite elements mean the elements made with virtually small material constants. It can be considered that the layer of virtual finite elements does not influence to the static/dynamic response of the structure. The analysis of the structure is carried out. For dynamic characteristic analysis, the natural frequencies and natural modes will be computed. For static and forced vibration analysis, the deformation of the structure due to external forces will be computed. Using the results, various parameters to be used for the optimization are computed. For dynamic analysis, stress energy and kinetic energy of each finite element will be computed as shown respectively below:

$$S_i^e = \frac{\{\phi_i^e\}^{\mathrm{T}}[\mathrm{K}_i^e]\{\phi_i^e\}}{2} \qquad \text{...(1)}$$

$$T_i^e = \frac{\{\phi_i^e\}^{\mathrm{T}}[M_i^e]\{\phi_i^e\}}{2} \qquad \text{...(2)}$$

where $\{\phi_i^e\}$ represents the partial mode vector corresponding to the freedom belonging to the finite element No.i. The eigenvalue can be obtained by Eq.(3) as the Rayleigh quotient with

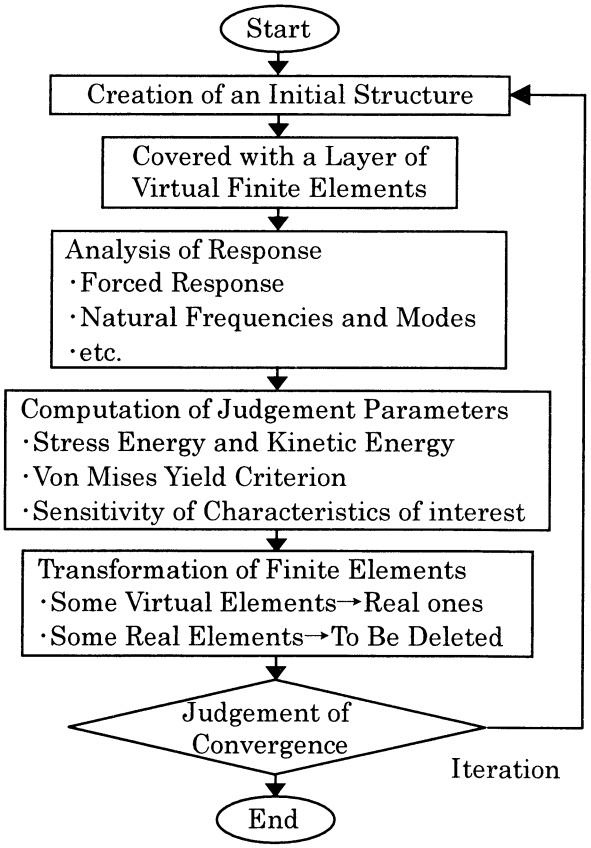


Fig.1 Flowchart of Algorithm

the total stress energy and the kinetic energy stored in the structure.

$$\lambda = \frac{S}{T} = \frac{\sum S_i^e}{\sum T_i^e} \qquad ...(3)$$

Therefore, an approximate sensitivity of the eigenvalue with respect to the finite element No.i can be formulated by Eq.(4).

$$\Delta\lambda_i = \frac{S - S_i^e}{T - T_i^e} - \lambda \qquad ...(4)$$

For static or damage analysis, a criterion should be adopted and computed. In this paper, Von Mises yield criterion is adopted. According to those parameters, appropriate transformation of finite elements is carried out. In this process, some virtual finite elements will be transformed to real finite elements. Some real finite elements will be deleted if they are judged to be unnecessary. Finally, the global objective function to be minimized or maximized is evaluated. The above-mentioned sequential processes are iterated until the objective function is converged.

3. Simple demonstrations

3.1 Static Optimizations

Figure 2 shows an optimization of a cantilever by applying a static force. In this case, the feasible design area is not limited to the upward and downward directions. The model of the structure consists of cubic finite elements whose sizes are all identically 5mmx5mmx5mm. The material is assumed to be iron. The length of the initial structure (cantilever) is 200mm. The size of the cross-section is 10mmx10 mm. Figure 3 shows how many finite elements are used for the structural model on every iterative stage. Figure 4 shows another optimization of the same structural model subject to a limitation of feasible design area, which is represented by the rectangular frame together with the initial structure. The optimized shape is different from the previous result due to the limitation of design area.

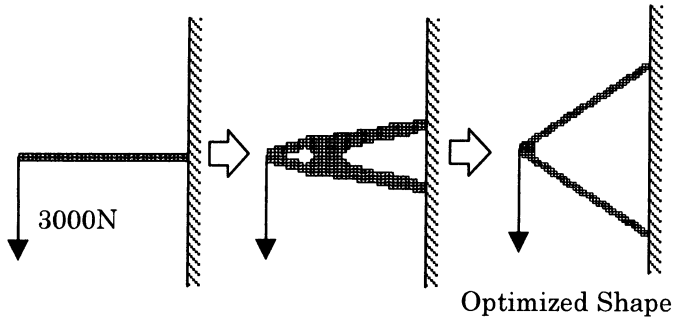


Fig.2 Static Optimization without limitation of design area upward and downward

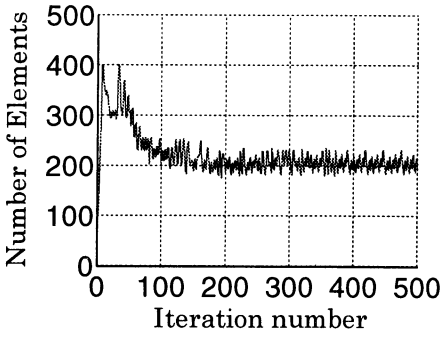


Fig. 3 Change of the Number of Finite Elements

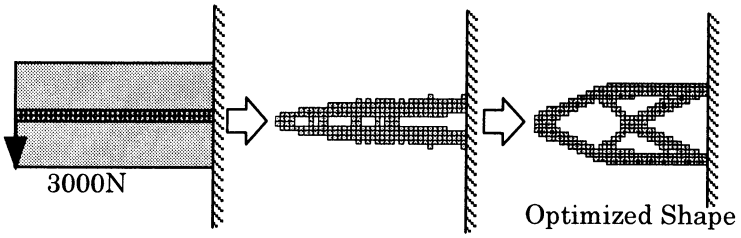


Fig. 4 Static Optimization with Limitation of Design Area

3.2 Dynamic Optimizations

Figure 5 shows an optimization of the same cantilever. In this case, the objective function is its first natural frequency. The cantilever is optimized in order to shift it up as high as possible. The constraint is the maximum number of finite elements available to use, which is 300. Figure 6 shows how its first natural frequency moved up. The first natural frequency of the initial structure is lower than 500 Hz. It is successfully shifted up to approximately 1500 Hz by the optimization.

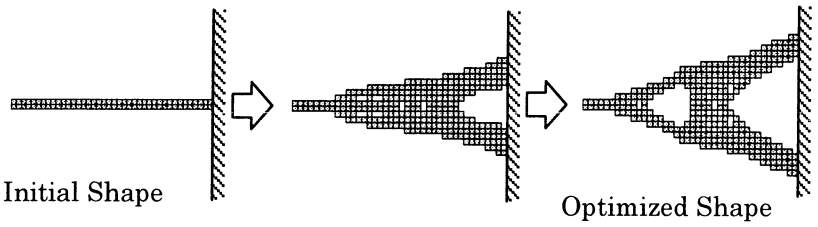


Fig. 5 Optimization for Shifting Up Its First Natural Frequency as High as Possible

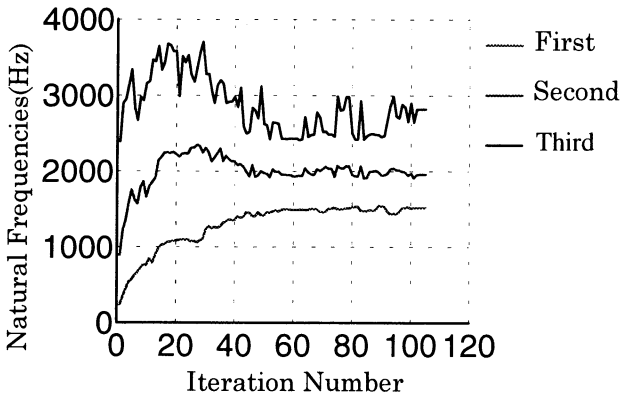


Fig.6 Change of Natural Frequencies

4. Conclusions

The authors presented a new evolutionary optimization method for structural dynamics and statics. In this method, the surface of a structural model to be optimized is covered with a layer of the virtual finite elements. At every iterative computation the structure evolves. Therefore, the optimization can easily start with a simple and small initial structure. The theory of modal analysis contributes for practical efficiency of computation regarding dynamic optimization. Simple optimizations of a 2-D model were presented to demonstrate clearly the basic practicability of the method.

Acknowledgement

This research work was carried out with the support of Grant-in-Aid for COE Research Project supported by the Ministry of Education, Science, Sports and Culture. The authors are grateful for this support.

References

[1] M.P.Bendsoe and N.Kikuchi (1988), Generating Optimal Topologies in Structural Design using a Homogenization Method, Computer Methods in Applied Mechanics and Engineering 71;197-224

[2] N.Inou, T.Uesugi, K.Todoroki and S.Ujihashi (1999), Self-Organization of Topological Structures by a Cellular Automaton, Synthesis in Bio Solid Mechanics, KLUWER ACADEMIC PUBLISHERS;21-32

[3] Y.Umetani and S.Hirai (1975), An Adaptive Shape Optimization Method for Structual Material Using the Growing-Reforming Procedure, Joint JSME-ASME Applied Mechanics Western Conference;359-365

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A Circularly Linked Synergetic Neural Networks

Masahiro Nakagawa

Department of Electrical Engineering,
Faculty of Engineering, Nagaoka University of Technology,
Kamitomioka 1603-1, Nagaoka, Niigata 940-2188, Japan
masanaka@vos.nagaokaut.ac.jp

Abstract - In this study we shall put forward a circularly connected synergetic neural network extending the previously studied auto-correlation or cross-correlation dynamics so as to realise the group memory retrieval. The present model is substantially based on a top-down formulation of the dynamic rule of an analog neural network in the similar manner to the conventional synergetic dynamics proposed by Haken. It is proved that a complete association can be assured up to the same number of the embedded patterns as the minimal number of neurons of the linked synergetic neural networks.

1. INTRODUCTION

Since the pioneering works by Anderson[1], Kohonen[2], and Nakano[3], many researchers have theoretically analysed such association models of the connected neurons with certain autocorrelation matrices (Amari[4], Amit et al.[5]) and Gardner[6]). Up to date, however, it has been reported that the relative memory capacity of the autocorrelation associative model is estimated as ~0.15N at most where N is the number of neurons. This has been considered as the result of the many spurious memory states increasing exponentially with the increase of N. In addition , as is well known nowadays, the performance is usually degraded for strongly correlated (or non-orthogonal) embedded patterns (L). To avoid this trapping problem, a few models of chaotic neural networks have been investigated recently (Aihara et al.[7,8], Nakagawa and Okabe[9], Nakamura and Nakagawa[10]). Therein chaotic behaviour has been shown to be applicable to the optimization problems. Moreover a novel type of chaotic neuron model has been proposed by the present author (Nakagawa[11-18]) and applied to some practical problems. (Kasahara and Nakagawa[19,20], Okada and Nakagawa[21]). Unfortunately, however, there had been no report of a model which assures the completeness of the memory retrievals.

In another aspect, a synergetic computer was proposed by Haken in the framework of the top-down approach based on cooperative phenomena, i.e. synergetics within the framework of the time differential form of the steepest descent dynamics of the energy (Fuchs and Haken [22-24], Haken [25]). Therein, however, their framework has been restricted to binary embedded patterns and a deterministic model with a continuous time, which was proved to assure the completeness of the memory retrievals. Recently the present author has put forward an association model with the discrete time synergetic dynamics, which is considered to be applicable for the retrieval problems such that L≤N, on the basis of the synergetics (Nakagawa[26,27]). Instead of the above-noted deterministic synergetic neural networks with continuous time (Haken[25]) , a stochastic discrete time model (Nakagawa[26,27]) had been also proposed for the time evolution of the neuro-system and applied to a practical problem of association. The stochastic property was found to be accompanied with fractal properties, in their dynamic behaviours of the order parameters, which were straightforwardly evaluated by means of the critical exponent method.(Nakagawa[28]) Recently a unified theory of chaos neural network and the synergetic approach has been reported (Nakagawa[29,30]) combining the synergetic approach (Nakagawa[27]) and the periodic chaos neural network (Nakagawa[11-18]). Furthermore, very recently, an autonomous chaos neuron model for the periodic chaos neural network has been also put forward by the present author (Nakagawa[31]). Therein the complete association was found to be assured up to L/N=1 and the initial directional cosine ~0.6. Even in a searching mode without any key information to find an embedded pattern, one may realise the complete association up to L/N~0.7. In these investigations concerned with the associative memories, some advantages of the chaotic approaches have been claimed in comparison with nonchaotic ones.

In another respect of the association model, Kosko has proposed a bidirectional associative memory (Kosko[32]) introducing a crosscorrelation matrix as the weighting connections between a couple of neuron layers, which may be classified two groups as reference and data. Up to date, however, there has been no report concerned with an extension of the bidirectional associative memory involving the synergetic dynamics which may assure the complete associations as were confirmed in the autoassociation models. (Haken[25], Nakagawa[26,27]) From this view point, in the very recent work, we constructed a bidirectional synergetic model (Nakagawa [33]) generalising the previous model (Nakagawa[27]). In that model, a complete association of a paired embedded patterns was found to be attained in an order parameter space. The uniqueness of the associated pattern was also examined from the viewpoint of the

stability analysis of a fixed point.

In this paper, we shall further generalise the previously argued synergetic models (Nakagawa[27]), Nakagawa[33]) so as to realise one to many or many to one association as well as a group association based on the circularly connected synergetic dynamics which assures the complete association.

2. THEORY

In the present model, we shall focus our interests on a circularly connected synergetic neural networks (See Fig.1.), in which the outputs of the i th neuron are denoted as $p^{(\alpha)}{}_i$ $(1\le i \le M^{(\alpha)}, 1\le\alpha\le G)$, whereM $^{(\alpha)}$ and G are the dimension of the αth network and the number of the circularly connected networks. With the discrete time n, the present analogue neuron model is defined by,

$$p^{(\alpha)}{}_i(n+1)=p^{(\alpha)}{}_i(n) - (1-k)\frac{\partial E}{\partial p^{(\alpha)}{}_i{}^{\dagger}(n)} \quad (1\le i \le M^{(\alpha)},\ 1\le\alpha\le G), \tag{1}$$

where k is called the acceleration parameter and ranges between 0 and 1, $p^{(\alpha)}{}_i{}^{\dagger}(n)$ $(1\le i \le M^{(\alpha)}, 1\le\alpha\le G)$ are covariant vectors for $p^{(\alpha)}{}_i$ $(1\le i \le M^{(\alpha)},\ 1\le\alpha\le G)$ defined by

$$p^{(\alpha)\dagger}{}_i(n)=\sum_{r=1}^{L}\sum_{j=1}^{M^{(\alpha)}} u^{(r,\alpha)\dagger}{}_i u^{(r,\alpha)\dagger}{}_j p^{(\alpha)}{}_j(n) \quad (1\le i\le M^{(\alpha)},\ 1\le\alpha\le G); \tag{2}$$

here L is the number of the embedded patterns for each network, $u^{(r,\alpha)\dagger}{}_i$ $(1\le r\le L, 1\le i\le M^{(\alpha)}, 1\le\alpha\le G)$ are the conjugate vectors, or the generalised inverse matrices, (e.g. Gantmacher[34], or Kohonen & Ruohonen[35]) of the embedded vectors as $u^{(r,\alpha)}{}_i$ $(1\le r\le L, 1\le i\le M^{(\alpha)}, 1\le\alpha\le G)$. Then the energy E is assumed to be given by

$$\begin{aligned} E= & -\sum_{r=1}^{L}\sum_{\alpha=1}^{G}\lambda^{(r,\alpha,\alpha+1)}(n)\langle u^{(r,\alpha)\dagger}, p^{(\alpha)}(n)\rangle \\ & \cdot\langle u^{(r,\alpha+1)\dagger}, p^{(\alpha+1)}(n)\rangle \\ & +\frac{1}{2}\sum_{r\neq r'}^{L}\sum_{\alpha=1}^{G}\langle u^{(r,\alpha)\dagger}, p^{(\alpha)}(n)\rangle^2 \\ & \cdot\langle u^{(r',\alpha+1)\dagger}, p^{(\alpha+1)}(n)\rangle^2 \\ & +\frac{1}{2}\sum_{r=1}^{L}\sum_{\alpha=1}^{G}\langle p^{(\alpha)\dagger}(n), p^{(\alpha)}(n)\rangle \\ & \cdot\langle p^{(\alpha+1)\dagger}(n), p^{(\alpha+1)}(n)\rangle \end{aligned} \tag{3}$$

where it is assumed that $\mathbf{p}^{(G+1)}=\mathbf{p}^{(1)}$, $\mathbf{u}^{(G+1)}=\mathbf{u}^{(1)}$ and $\mathbf{u}^{(G+1)\dagger}=\mathbf{u}^{(1)\dagger}$, which means our synergetic neural network is circularly connected, and the symbol $\langle\ ,\ \rangle$ stands for the innerproduct, $\lambda^{(r,\alpha,\alpha+1)}(n)$ is the attraction parameter of the r th pattern between αth network and $(\alpha+1)$th one, and takes a positive value for the attraction to the embedded pattern and otherwise is set to 0. (e.g.Haken[25]) It should be noted here that we have the following orthogonal relation between the covariant vectors $u^{(r,\alpha)\dagger}{}_i$ and the contravariant ones, $u^{(r,\alpha)}{}_i$;

$$\sum_{i=1}^{M^{(\alpha)}} u^{(r,\alpha)\dagger}{}_i u^{(s,\alpha)}{}_i=\delta_{rs}. \tag{4}$$

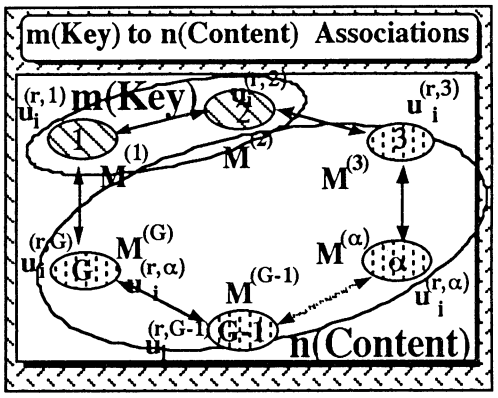


Fig. 1 Circularly connected Synergetic Neural Nets (SNN)

Substituting Eq. (3) into Eq.(1), one readily has

$$p^{(\alpha)}{}_i(n+1)=kp^{(\alpha)}{}_i(n)+(1-k)p^{*(\alpha)}{}_i(n), \tag{5}$$

$$\begin{aligned} p^{*(\alpha)}{}_i(n)= & p^{(\alpha)}{}_i(n) \\ & +\sum_{r=1}^{L}\Big\{\lambda^{(r,\alpha-1,\alpha)}(n)\langle u^{(r,\alpha-1)\dagger}, p^{(\alpha-1)}(n)\rangle \\ & +\lambda^{(r,\alpha,\alpha+1)}(n)\langle u^{(r,\alpha+1)\dagger}, p^{(\alpha+1)}(n)\rangle\Big\}u^{(r,\alpha)}{}_i \\ & -\sum_{r\neq r'}^{L}\sum^{L}\Big\{\langle u^{(r,\alpha-1)\dagger}, p^{(\alpha-1)}(n)\rangle^2 \\ & +\langle u^{(r,\alpha+1)\dagger}, p^{(\alpha+1)}(n)\rangle^2\Big\} \\ & \cdot\langle u^{(r',\alpha)\dagger}, p^{(\alpha)}(n)\rangle u^{(r',\alpha)}{}_i \\ & -\Big\{\langle p^{(\alpha-1)\dagger}(n), p^{(\alpha-1)}(n)\rangle \\ & +\langle p^{(\alpha+1)\dagger}(n), p^{(\alpha+1)}(n)\rangle\Big\}p^{(\alpha)}{}_i(n) \end{aligned} \tag{6}$$

The state of the i th neuron in the αth network, $p^{(\alpha)}{}_i(n)$, at time n may be expressed in terms of the linear combination of the embedded pattern vectors $u^{(r,\alpha)}{}_i$ $(1\le i\le M^{(\alpha)}, 1\le r\le L)$ as follows

$$p^{(\alpha)}{}_i(n)=\sum_{r=1}^{L} m^{(r,\alpha)}(n)u^{(r,\alpha)}{}_i+\mu^{(\alpha)}{}_i(n) \quad (1\le i\le M^{(\alpha)},\ 1\le\alpha\le G), \tag{7}$$

$$m^{(r,\alpha)}(n)=\sum_{i=1}^{M^{(\alpha)}} u^{(r,\alpha)\dagger}{}_i p^{(\alpha)}{}_i(n) = \langle \mathbf{u}^{(r,\alpha)\dagger}, \mathbf{p}^{(\alpha)}(n)\rangle \quad (8)$$

where $\mu^{(\alpha)}{}_i(n)$ $(1\le i\le M^{(\alpha)},\ 1\le\alpha\le G)$ are the residual errors or noises due to the incompleteness of the vector set $u^{(r,\alpha)}{}_i$.

The definition of the covariant vectors can be expressed as

$$p^{(\alpha)\dagger}{}_i(n)=\sum_{r=1}^{L}\sum_{j=1}^{M^{(\alpha)}} u^{(r,\alpha)\dagger}{}_i u^{(r,\alpha)\dagger}{}_j p^{(\alpha)}{}_j(n) \quad (9)$$

On the contrary, the contravariant vectors are to be given by

$$p^{(\alpha)}{}_i(n)=\sum_{r=1}^{L}\sum_{j=1}^{M^{(\alpha)}} u^{(r,\alpha)}{}_i u^{(r,\alpha)}{}_j p^{(\alpha)\dagger}{}_j(n) \quad (10)$$

Then Eq. (6) reads

$$\begin{aligned} p^{*(\alpha)}{}_i(n)&=p^{(\alpha)}{}_i(n)\\ &+\sum_{r=1}^{L}\left\{\lambda^{(r,\alpha-1,\alpha)}(n)m^{(r,\alpha-1)}(n)\right.\\ &\left.+\lambda^{(r,s,\alpha+1)}(n)m^{(r,\alpha+1)}(n)\right\}u^{(r,\alpha)}{}_i\\ &-\sum_{r\neq r'}^{L}\sum^{L}\left(m^{(r,\alpha-1)}(n)^2+m^{(r,\alpha+1)}(n)^2\right)\\ &\quad\cdot m^{(r',\alpha)}(n)u^{(r',\alpha)}{}_i\\ &-\left\{\langle p^{(\alpha-1)\dagger}(n),\ p^{(\alpha-1)}(n)\rangle\right.\\ &\left.+\langle p^{(\alpha+1)\dagger}(n),\ p^{(\alpha+1)}(n)\rangle\right\}p^{(\alpha)}{}_i(n) \end{aligned} \quad (11)$$

Now $\mu^{(\alpha)}{}_i(n)$ can be expanded as

$$\mu^{(\alpha)}{}_i(n)=\sum_{r=1}^{M^{(\alpha)}-L}\alpha^{(r,\alpha)}(n)u^{*(r,\alpha)}{}_i, \quad (12)$$

where the vector set, $u^{*(r,\alpha)}{}_i$, can be chosen to satisfy the following orthogonal relations (Haken[25]) with respect to $u^{(r,\alpha)}{}_i$. Then the innerproducts ofu $u^{(r,\alpha)\dagger}{}_i$ and also $u^{*(r,\alpha)\dagger}{}_i$ with Eq.(5) yields

$$\begin{aligned} m^{(r,\alpha)}(n+1)&=m^{(r,\alpha)}(n)\\ &+(1-k)\left\{\left\{\lambda^{(r,\alpha-1,\alpha)}(n)\,m^{(r,\alpha-1)}(n)\right.\right.\\ &\left.\left.+\lambda^{(r,\alpha,\alpha+1)}(n)\,m^{(r,\alpha+1)}(n)\right\}\right.\\ &\left.-\sum_{r'\neq r}^{L}\left(m^{(r',\alpha-1)}(n)^2+m^{(r',\alpha+1)}(n)^2\right)m^{(r,\alpha)}(n)\right.\\ &\left.-\left\{\sum_{s=1}^{L}m^{(s,\alpha-1)}(n)^2+\sum_{s=1}^{M^{(\alpha-1)}-L}\alpha^{(s,\alpha-1)}(n)^2\right.\right.\\ &\left.\left.+\sum_{s=1}^{L}m^{(s,\alpha+1)}(n)^2+\sum_{s=1}^{M^{(\alpha+1)}-L}\alpha^{(s,\alpha+1)}(n)^2\right\}\right.\\ &\left.\cdot m^{(r,\alpha)}(n)\right\} \end{aligned} \quad (13)$$

and

$$\begin{aligned} \alpha^{(r,\alpha)}(n+1)&=\alpha^{(r,\alpha)}(n)\\ &-(1-k)\left\{\sum_{s=1}^{L}m^{(s,\alpha-1)}(n)^2+\sum_{s=1}^{M^{(\alpha-1)}-L}\alpha^{(s,\alpha-1)}(n)^2\right.\\ &\left.+\sum_{s=1}^{L}m^{(s,\alpha+1)}(n)^2+\sum_{s=1}^{M^{(\alpha+1)}-L}\alpha^{(s,\alpha+1)}(n)^2\right\}\\ &\cdot\alpha^{(r,\alpha)}(n) \end{aligned} \quad (14)$$

3. RESULTS

In this section, let us present a numerical result example concerned with the searching mode, which are realised by means of the control of the attraction parameters, $\lambda^{(r,\alpha,\alpha+1)}$ $(1\le\alpha\le G,\ 1\le r\le L)$.

Here we shall set the parameters as L=10, G=3 $M^{(1)}=40$, $M^{(2)}=60$, $M^{(3)}=80$, k=0.8 if not mentioned hereafter, for numerical computations. Then the embedded vectors, $u^{(r,\alpha)}{}_i$ $(1\le\alpha\le G)$, were set to random vectors, which range over {-1~+1} in terms of the following definitions,

$$u^{(r,\alpha)}{}_i=2\,\mathbf{z}-1 \quad ,(1\le\alpha\le G)(1\le r\le L)(1\le i\le M) \quad (15)$$

where **z** stands for the pseudo-random numbers ranging over 0~1.

In this report let us present the results for memory searching mode as seen in the previous works (Haken [25], Nakagawa[26,27]). The corresponding result is given in Fig.2. One may confirm that the memory searching of pairs of the order parameters, $m^{(s,\alpha)}(n)$ $(1\le\alpha\le G)$, can be accomplished under this dynamic retrieval mode.

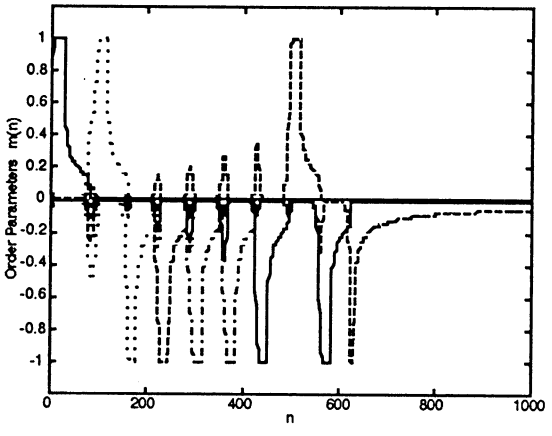


Fig.2 The time dependence of $m^{(r)}(n)=m^{(r,1)}(n)$

REFERENCES

[1]Anderson, J.A. (1972). "A simple neural network

generating interactive memory," Mathematical Biosciences, **14**,197-220.
[2]Kohonen, T.(1972). "Correlation matrix memories," IEEE Transaction on Computers, **C-21**,353-359.
[3]Nakano, K.(1972). Association. IEEE Transactions on System, Man, and Cybernetics, **SMC-2**,380-388.
[4]Amari, S. (1977). Neural Theory of association and concept formation. Biological Cybernetics, **26**,175-185.
[5]Amit, D.J. , Gutfreund, H. and Sompolinsky, H.(1985). "Storing infinite numbers of patterns in a spin-glass model of neural networks," Physical Review Letters,**55**,1530-1533.
[6]Gardner, E.(1986). "Structure of metastable states in the Hopfield model," Journal of Physics, **A19**,L1047-1052.
[7]Aihara, K. and Matsumoto, G.(1986). "Forced oscillations and routes to chaos in the Hodgkin-Huxley axons and squid giant axons, in Chaos in Biological Systems"(eds. Degn, H. , Holden A. V. and Olsen L.F., Plenum Press)121-131.
[8]Aihara, K., Numajiri, T., Matsumoto, G. and Kotani, M.(1986). "Structures of attractors in periodically forced neural oscillators". Physics Letters A**116**,313-317.
[9]Nakagawa, M. and Okabe, M.(1992). "On the chaos region of the modified Nagumo-Sato model," Journal Physical Society of Japan, **61**,L1121-1124.
[10]Nakamura, K and Nakagawa, M.(1993). " On the associative model with parameter controlled chaos neurons," Journal Physical Society of Japan,**62**,2942-2955.
[11]Nakagawa, M.(1994). "A model of chaos neuro-dynamics with a periodic activation function," Proc. of ICONIP'94 (Seoul),609-613.
[12]Nakagawa, M.(1995). "An artificial neuron model with a periodic activation function," Journal of Physical.Society of.Japan, **64**,1023-1031.
[13]Nakagawa, M.(1995). " A model of periodic chaos neuro-dynamics," Proceedings of International Conference on Dynamical Systems and Chaos, **2**,603-607.
[14]Nakagawa, M.(1995). "A study of chaos neural networks with a periodic activation function" Proc. of International Conference of Artificial Neural Networks and Expert Systems'95(New Zealand)45-50.
[15]Nakagawa, M.(1998). "Chaos associative memory with a periodic activation function" Journal of Physical Society of Japan, **67**,No.7, 2281-2293.
[16]Nakagawa, M.(1999). " A super memory retrieval with chaos associative model" Journal of Physical Society of Japan, **68**(in press).
[17]Nakagawa, M.(1999). " *Chaos and Fractals in Engineering* "World Scientific Inc.
[18]Nakagawa, M.(1999). " A Chaos Associative Model with a Sinusoidal Activation Function ", Chaos, Solitons and Fractals, **10**, No.9, 1437-1452.
[19]Kasahara, T. and Nakagawa, M. (1995). "On the parameter controlled chaos neural network," Electronics and Communications in Japan Part III-Fundamentals,**78**-7,1-10.
[20]Kasahara, T. and Nakagawa, M. (1995). " A study of association model with periodic chaos neurons," Journal of Physical Society of Japan,**64**,4964-4977.
[21]Okada and Nakagawa(1999). " A Study of Back Propagation Learning with Periodic Chaos Neurons", Chaos, Solitons and Fractals, **10**, No.1, 77-97.
[22]Fuchs, A. and Haken, H. (1988). "Pattern Recognition and Associative Memory as Dynamical Processes in a Synergetic System I," Biological Cybernetics,**60**,17-22.
[23]Fuchs, A. and Haken, H.(1988). "Pattern Recognition and Associative Memory as Dynamical Processes in a Synergetic System II," Biological Cybernetics,**60**,107-109.
[24]Fuchs, A. and Haken, H.(1988). " *In Dynamic Patterns in Complex Systems* " ed. by Kelso, J.A.S. , Mandell, A.J.and Shlesinger, M. F. (World Scientific, Singapore).
[25]Haken, H.(1991). "*Synergetic Computers and Cognition* "(Springer-Verlag).
[26]Nakagawa, M.(1993). "A study of association model based on synergetics," Proceedings of International Joint Conference on Neural Networks '93 NAGOYA, JAPAN,2367-2370.
[27]Nakagawa, M.(1995). "A synergetic neural network" IEICE Fundamentals E7**8-A**,412-423.
[28]Nakagawa, M.(1993). "A critical exponent method to evaluate fractal dimensions of self-affine data" Journal Physical Society of Japan, **62**,4233-4239.
[29]Nakagawa, M.(1995). "A chaos synergetic neural network" Journal of Physical Society of Japan, **64**,1023-1031.
[30]Nakagawa, M. (1995). "Periodic chaos neural networks," Proc. of International Conference of Neural Networks'95(Australia),**6**,3028-3033.
[31]Nakagawa, M.(1996). "A parameter controlled chaos neural network" Journal of Physical Society of Japan, **65**,No.6, 1859-1867.
[32]Kosko, B. (1988). " Bidirectional associative memories," IEEE Transaction SMC, **18**, 49-60.
[33]Nakagawa, M.(1997). "A synergetic neural network with crosscorrelation dynamics" IEICE Fundamentals E, **65**,No.6, 1859-1867.
[34]Gantmacher, F.R.(1959). "*The Theory of Matrices* " (Chelsea Publishing Company).
[35]Kohonen, T. and Ruohonen, M.(1973). "Representation of associated pairs by matrix operators," IEEE Trans., **C-22**, 701-702.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Neural Network Human-Face Recognition Using Wavelet Mosaic Pattern

Lifeng ZHANG
Electrical Engineering Deparment
Kyushu Institute of Technology
Kitakyushu City, 804-8550, Japan

Hiroshi KONDO
Electrical Engineering Deparment
Kyushu Institute of Technology
Kitakyushu City, 804-8550, Japan

Abstract

A robust human-face recognition using neural network with a wavelet mosaic pattern is presented. The wavelet transform details include the special feature of the image. In this paper the approximation of the wavelet transform detail is utilized as an input of the neural network. In a human-face recognition a difference of the face expressions makes it difficult. Appropriate level approximation pattern of the discrete wavelet transform(DWT) details, however, keeps the fundamental feature of the face without depending a little difference of the expression. As a result of employing such patterns the robustness property of the back propagation neural network becomes very large and the neural network which is not so sensitive to human expressions can be constructed. Many simulation examples show that the presented neural network works so well for different face expressions, i.e., smiling, laughing, irritating, touching, etc.

key words *Neural Network, Human-Face Recognition, Wavelet Mosaic*

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 because of its fast computing time. The neural network, however, usually works well only for a predetermined front face. In the case where the object face is a smiling or a crying or an irritating one the neural system has not worked well [1]~[6]. Furthermore, it has not worked well also for an up facing, or a down facing, or a right or left-rotated face. This is because each of these faces has a different face expression with a serious face, which is used as a teacher ' s pattern of the neural network. Hence if we employ all of such face as teacher ' s patterns for recognition, then maybe the recognition will be attained completely. But our human being has so many different expressions and then the neural network must have too many teachers ' patterns. It is impossible to have such many faces as teacher ' s patterns for the network and furthermore even if it can be done, it looks that the convergence of the network is very hard. In order to solve such problems here a wavelet mosaic face pattern is utilized. 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 [3]. We utilize such mosaic patterns of wavelet transform details as an input for the neural network with three layers. Wavelet transform details can be utilized well for detecting the special feature of the face image. Hence the DWT approximation of the detail is applied in order to make a mosaic pattern. Typical faces with different expressions are applied for such mosaic patterns here. The mosaic pattern needs only few typical faces because of its robustness nature. There is an excellent paper [4] for face recognition using a mosaic pattern. The paper, however, treats only the front face with the size 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 many different face expressions but also the rotated one.

2 Back Propagation Neural Network

The normatization of the input face gray level must be in order to reduce the variation of the input face data due to something like a different lighting. The region of the face image for this neural network input data includes the hair. The number of the input layer neurons of the neural network is 8880 ($30 \times 37 \times 8$) and that of the output layer is 41 (40 different human-face +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 50 persons faces except the 40 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 45 which is determined by using integer dynamic programming. Although we introduce one more output neuron for unfamiliar or strange 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 method ash been preferable to the latter one in our many simulations. The error rate of the latter system is 15

3 DWT Mosaic Pattern

Discrete wavelet transform has some interesting properties for digital signal processing [7][8][9]. In this paper discrete wavelet transform (DWT) is employed for the input image of the neural network. DWT is expressed as

$$d_k^{(j)} = 2^j \int_{-\infty}^{\infty} \overline{\varphi(2^j x - k)} f(x) dx \qquad (1)$$

$$(j,\ k : \text{integer})$$

where $d_k^{(j)}$ is a DWT coefficient and $\varphi(2^j x - k)$ is called mother wavelet function. We employ a Doubechies 1 as a mother wavelet function because it is simple for calculation and sufficient for our purpose. Here using DWT of the face image we perform the multiple level analysis. Figure 1 shows the first level analysis of DWT. The right upper is called a horizontal detail, left lower is a vertical one and the right lower is a diagonal one. The vertical and horizontal details have an enhanced feature of the face as shown in Fig. 1. The left upper is called an approximation, which includes low frequency components of the original and looks like the original image thanks to the down sampling.

These all images are half sized of the original because of the down sampling. We want to use the horizontal and vertical details for our neural network input. However a neural network is very sensitive to an image shift. Hence the mosaic pattern of the above images is preferable. But taking a mosaic pattern directly from the DWT detail gives sometimes a lack of feature-image. Hence we utilize here the DWT approximation of details because DWT approximation has a mild averaging property of the original image. Fig.2a and 2b are DWT analysis(approximation) of the horizontal and vertical details of Fig.1. Fig.2c is a mixed

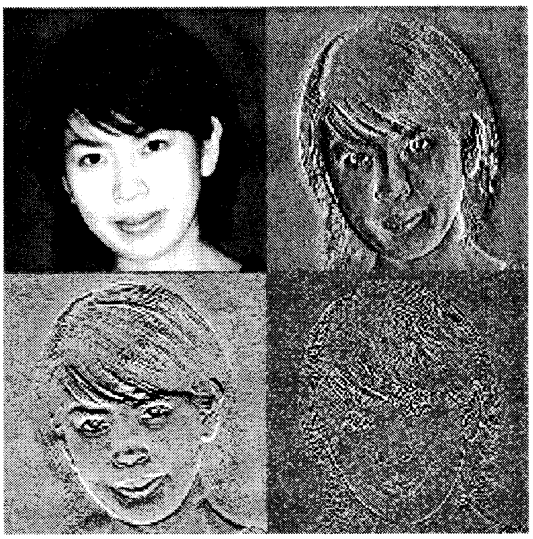


Figure 1: Discrete wavelet transform (level 1)



a *b* *c*

Figure 2: a. Approximation of the horizontal detail. b.Approximation of the vertical detail. c. Mixed image of a and b

image of Fig.2a and 2b made by a geometrical mean. We call this a wavelet mosaic pattern. The wavelet mosaic pattern has still a detailed feature of the face but has also a robustness property for the image shift. This means that it is one of the best input ways of a neural network.

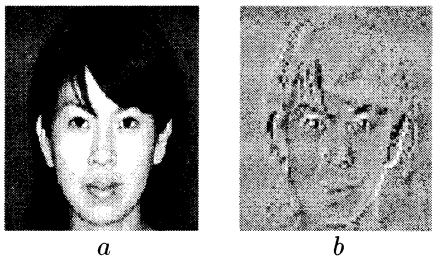
4 Instruction Patterns

Usually the straight (serious) front face is taken as a teacher's(or instruction) pattern. But actually we have a lot of face expressions. Hence in face recognition problems one of most important things is "how to choose the teacher ' s patterns". It should be determined from the experimental results which face of many different expressions should be selected as a teacher ' s pattern for the neural network except the straight (serious) front face. In this paper we utilize a mosaic pattern which is able to detect the common feature of the face among the somewhat different ex-

pressions. For example, Fig. 3a and Fig. 4a show a serious face and a smile one and Fig. 3b and Fig. 4b are the mosaic patterns of them respectively. Figure 3b looks similar to Fig. 4b. In fact when the Fig. 4b is employed as an input test data for the back propagation neural network with a teacher ' s pattern of Fig. 3b the output react to the Fig. 3b completely. Figure 5d, however, which is a laughing face does not react to the Fig. 3. These are too much different expressions. Hence we employ the typical expressions for laughing, irritating, and crying faces. For example, as a typical laughing face we choose the face whose reaction value to the serious face teacher ' s pattern is less than 0.5 as shown in Table 1 and Fig. 5a Fig. 5d. Here, " smile 3 " is taken as a typical laughing face teacher ' s pattern. In this table face 2, face 3, ... are the other person ' s faces.

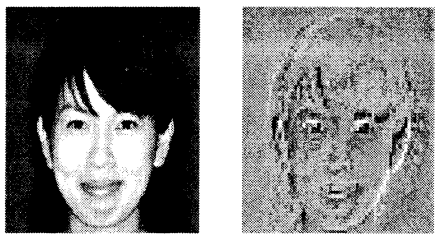
Teacher's pattern	Smile 1	Smile 2	Smile 3	Smile 4
Serious face 1	0.935	0.812	0.501	0.429
Serious face 2	0.222	0.123	0.080	0.211
Serious face 3	0.113	0.197	0.165	0.239
⋮				

Table 1 Reaction values for each smile face
(Teacher ' s patterns are 40 person ' s serious faces)



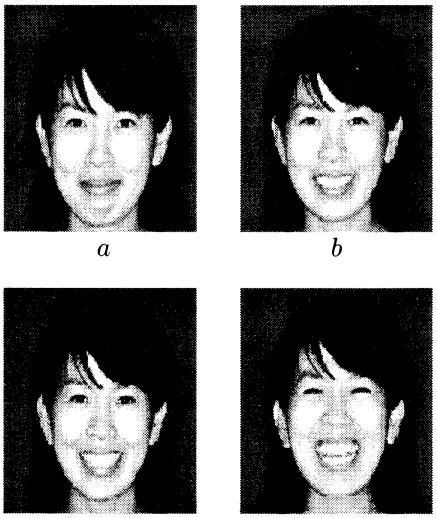
a *b*

Figure 3: a. Serious face. b. Wavelet Mosaic Pattern of a



a *b*

Figure 4: a. Smiling face. b. Wavelet Mosaic Pattern of a



a *b* *c* *d*

Figure 5: a Smile 1. b Smile 2. c Smile 3. d Smile 4.

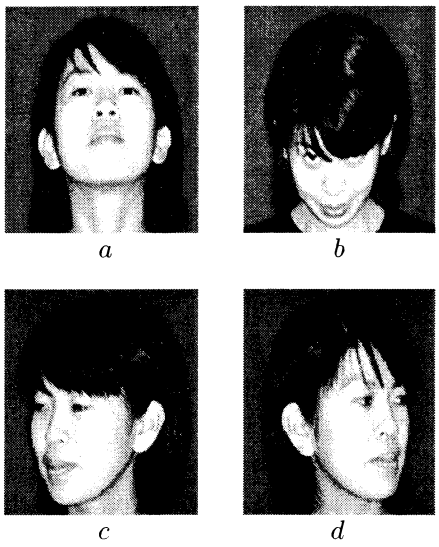


6 7

Figure 6: Crying face
Figure 7: Irritating face

Figures 6 and 7 are crying and irritating mosaic faces respectively. These images react to the serious or laughing image. Hence it is not needed to make such teacher ' s patterns. And Figs. 8a $\sim$ 8d are up looking, down looking, right rotated, and left rotated faces respectively. For these images we also determine the looking and rotated tolerance degree by checking the output value less than 0.5 for the neural network with straight serious face teacher ' s patterns. Consequently up looking degree for teacher ' s patter is 18° and the down looking degree is 17°. The right (left) rotation degree is 15°. Then it seems that the additional teacher ' s patterns are not also needed because of the pretty big tolerance.

Totally there are 2 mosaic teacher ' s patterns (serious and laughing) for each object person. The last output neuron is assigned to " others ". " Others " is made by 50 person ' s faces, which are different from object persons. That is, each wavelet mosaic patterns of 50 different " others " faces react to the same output neuron " others ". Making " others " output is



a b c d

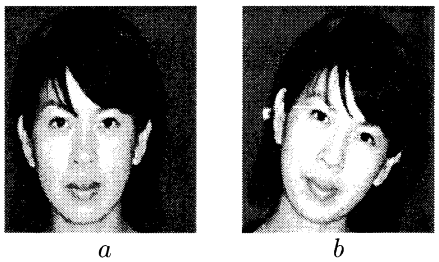
Figure 8: a Up looking face. b Down looking face. c Right rotated face. d Left rotated face.

very effective for such human face recognition. When a strange person 's face is input, the " others " output neuron will almost always react because, for example, the input mosaic face may have a lip similar to that of one of the 50 " others " faces, and the eyes similar to the different one of the 50 " others " faces, etc. Actually we have tested 100 strange persons to certify the above and got a recognition error for only one person. This person 's face is somewhat similar to that of one of the 40 object persons. Hence the output object neuron have reacted to it. The value of the output, however, is 0.613 and the " others " one is 0.498. If we take more than 50 persons as " others ", then the " others " neuron may react with a larger value than that of object neurons. Anyway, from this simulation test we confirm that making " others " output neuron is very effective for such neural network.

5 SIMULATION

First we have examined the robustness of the mosaic pattern to a shifted face image and a swing face image. Figure 9 is a right shifted version of an object image and Fig. 10 is a swing one.

Tables 2 and 3 show the output value for several sifted pixels of each direction for the neural network with only straight front faces. Since the right shift value is similar to the left one the values in Table 2 is taken as an average of the both. From these tables a mosaic pattern is very effective for making a robustness in a neural network in both swing and up



a b

Figure 9: Right Shift Image

Figure 10: Swing Image

shift, but we see only one or two pixels down shift version is recognizable. Up shift version values are not similar to those of down shift version because of the non-symmetric feature of a human face in vertical direction. These results show that even a mosaic pattern can not make a neural network robust so much especially in the downshift. Hence it is important to fix the center of gravity of the image. As for a swing head we can examine until 10 degrees from the vertical because the head part drop out from the image size 30 $\times$ 37 pixels.

Shift pixel	Output value
1	0.705
2	0.651
3	0.570
4	0.425
5	0.311
6	0.386
7	0.251

Table 2 Right (left) shift

Shift pixel	Output value	
	Up shift	Down shift
1	0.965	0.967
2	0.906	0.767
3	0.803	0.469
4	0.645	0.307
5	0.228	0.005
6	0.193	0.008
7	0.313	0.012
8	0.291	0.012
9	0.142	0.014
10	0.024	0.005
11	0.000	0.000

Table 3 Up and down shift

Rotation degree (clock wise)	Output value
1	0.918
2	0.779
3	0.862
4	0.824
5	0.689
6	0.512
7	0.347
8	0.318
9	0.267
10	0.191
11	0.164

Table 4 Swing head

Table 4 shows a result of the swing head image. Within 6 degrees of the swing test images we can recognize by the network with straight teacher ' s face patterns. Hence in real recognition case it seems that the correction of the main axis of the image is not needed in our network. The simulation tests are performed with 40 object persons with various different expressions plus 100 strange persons who are different from the " others " persons. Consequently, all object parsons has been recognized completely and in strange persons recognition only one parson has not been recognized, whose face is somewhat similar to that of one of the object persons. Totally the error rate is only 1 percent.

6 CONCLUSIONS

We have examined the tolerance property of the neural network with wavelet mosaic patterns. The simulation example shows that the presented neural network works so well for a face-swinging between plus or minus 6 degrees, right or left rotation 15 degrees, and up or down looking between plus or minus 17 degrees. Furthermore we also check the tolerance property for the shifted images and the different size images. For the image sift, the tolerance pixels are $2 \sim 3$ pixels. It means that a neural network is sometimes a little weak to such position sift. In order to overcome this defect we employ a matching way of the center of gravity of every image. Also for the size difference it should be needed to correct. It seems that such pre-processing or pre-correction must be needed for a neural network. Nevertheless a neural network has still a great merit for pattern recognition because of its quick response. We set the corresponding output unit value 0.9 and the other ones 0.1 for repeating study of the neural network. Now we have set it for recognition of 40 persons. In a real world, however, it looks that more than 100 persons should be applicable to such a system. We are now examining such a big system. Of cause for such many persons we can also take another way to run in parallel of the described system. These are the left problem to be solved soon. If the recognition for more than 40 persons are required, then we can make many systems of this that can run in parallel.

References

[1] L. Sirovich and M. Kirdy, " Low-dimensional procedure for the characterization of human faces, " J. Opt. Soc. Am. Vol. A-4, No.3, 1987 pp.519-524

[2] M. Turk and A. Pentland, "Face recognition using eigenfaces, " Proc. CVPR ' 91, June 1991

[3] L.D. Harmon and B. Julesz, " Masking in visual recognition: Effects of two-dimensional filtered noise, " Science, Vol. 180, No.15, 1973, pp.1194-1197

[4] M. Kosugi, "Human-face recognition using mosaic pattern and neural networks, " Trans. of IEICE, Vol. J76-D-II, No.6, June 1993 pp.1132-1139

[5] M.K. Fleming and G.W. Cotterll, " Categorization of face using unsupervised feature extraction," Proc. IJCNN90-SanDiego, II, 1990

[6] J.L. Perry, "Human face recognition using a multilayer perceptron," Proc. IJCNN90-WASH, II, 1990

[7] J.J. Bendetto and M.W. Franzier, " Wavelets-Mathematics and Applications", CRC Press, chapter 2, pp. 219-365, 1994

[8] A.Averbuch, D.Lazar, and M.Israeli, "Image compression Using Wavelet Transform and Multiresolution Decomposiyion", IEEE Trans. on image pro., vol.5, No.1, pp4-15, 1996

[9] W.Chen, S.Itoh, and J.Shiki, "Irregular Sampling Theory, Vol.44, No.3, pp1131-1142, 1998

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Evolutionary Neural Network Based on DNA Coding Method

Won-Hee Lee and Hoon Kang

School of Electrical & Electronics Engineering
College of Engineering, Chung-Ang University
221 Huksuk-dong, Dongjak-gu, Seoul 156-756, Korea
E-mail : wony@sirius.cie.cau.ac.kr

Abstract

Recently, neural networks are used in various fields of applications. They are specifically designed to meet a given problem, but if we change the target application, complex design processes are revisited. To avoid these difficulties, many neural network schemes utilizing the evolutionary methods are proposed and plausibly good results are revealed. However, we still suffer from redesign problems if changes are encountered. In this paper, we propose a new neural network evolving technique based on the DNA coding method. The initial population of the structure information and the weights for the neural network is generated, and then the descendants are chosen with fitness. The evolutionary technique and the suitable fitness measure are used to find a neural network with the fractal number of node, which represents a good result in a given application.

Keyword : Neural Network, Genetic Algorithm, DNA Coding Method.

1. Introduction

A neural network[1] is an information processing system that solves a complex problem using combination of simple processing elements. After 1980's, a neural network has been applied to a wide variety of problems such as storage and restoration of data or patterns, classifying patterns and mapping from patterns of inputs to outputs, and it is proven to be useful. For designing a neural network, there are some considerations, which is the number of layers, the number of nodes, learning algorithm, training data and etc. However, the reasonable methods of making up a neural network have not been proposed while the methods depending on experimental results does. To solve these problems, methods of constructing a neural network using genetic algorithm[2] have been proposed and showed good results[3][4].

In this paper, we applied the DNA coding method[5] to an automatic design and evolution of a neural network. The DNA coding method is modified genetic algorithm which was proposed by Yoshikawa in 1995.

2. Neural Network Model

In this paper, we assume some constraints in the neural network structure to evolve neural network using genetic algorithm.

- A neural network is restricted to feed-forward network.
- A neural network has layers but has no limitation to link.
- A hidden node has two inputs and one output.
- A hidden node is defined as three basic nodes; `A type' node, `O type' node and `N type' node.

this paper is supported by Brain Science Research Center of Korean Ministry of Science and Technology under the grant no. 98-J04-01-01-A-07

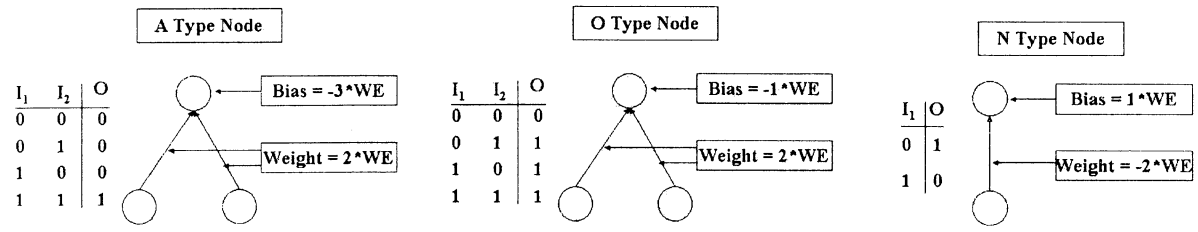


Figure 1. Basic Node (A type node, O type node, N type node)

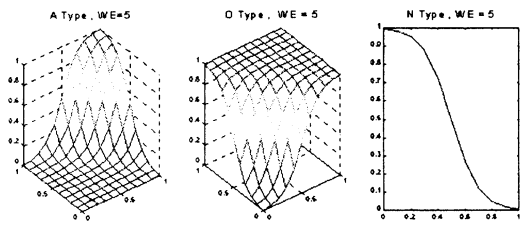


Figure 2. Output graph of Basic node (WE=5)

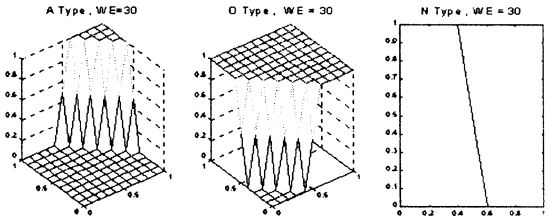


Figure 3. Output graph of Basic node (WE=30)

The each basic node consists of the fixed ratio of the weight to the bias. Therefore weights and bias don't change individually but just follow the element called the WE. Fig. 2, Fig. 3 show the characteristics of the basic nodes. The characteristics of the basic nodes with respect to input 0 and input 1 are in true table as figure 1. If the value of the WE is under 3, the characteristic of the WE doesn't follow the true table and If the WE value is over 30, the output of the node puts the same result out while the WE changes. If the WE value is negative number, `A type' node has the NAND output for input 0 and 1, `O type' node has the NOR output for input 0 and 1, and `N type' node has the output of the scaling for the input. Therefore the value of the WE can be limited from -30 to 30. The outputs of the neural network which consists of the basic nodes is calculated as same as the one of a general feed forward neural network.

$$net = \sum_{n=1}^{N} W_n * I_n$$

$$Out = \frac{1}{1+\exp(-(net+bias))}$$

W_n : weight of n`th input

I_n : output of n`th input node

'A type' and 'O type' node is N=2. 'N type' node is N=1.

3. Introduction DNA Coding method

Yoshikawa proposed a DNA coding method, a changed form of GA in 1995. It consists in Artificial DNA, using four symbols, such as A(adenine), G(guanine), C(cytosine) and T(thymine), not a binary representation which is used in GA. This chromosome contains genetic information. After this information is transformed and reproduced by crossover and mutation that are the basic operations of SGA(Simple Genetic Algorithm), it inherits to the next generation. Delivered information makes new individuals, and it propagate by their valued fitness in a Translation mechanism. In the DNA Coding Method, A genetic information depends on the order of codons in a gene, not the position of nucleotides in a chromosome.

A Codon is consist of three nucleotides in a DNA, and after translation it turn to Amino acid. So the total number of a codons is 64, in human 64 codons makes 20 amino acid. The arrangement of amino acid makes various protein, and it makes cell. A Gene in a Artificial DNA is in anywhere between Start Codon and Stop Codon, not in a fixed position. We can generate a variable number of genes in a same length DNA by including other Start Codon in a Gene.

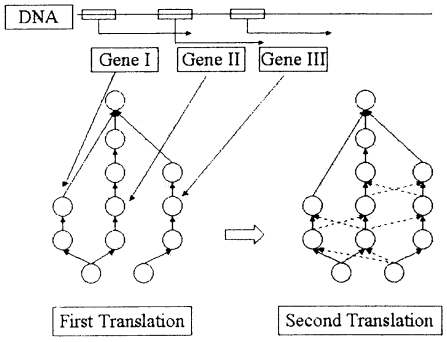


Figure 4. Construction Neural Network using DNA Code

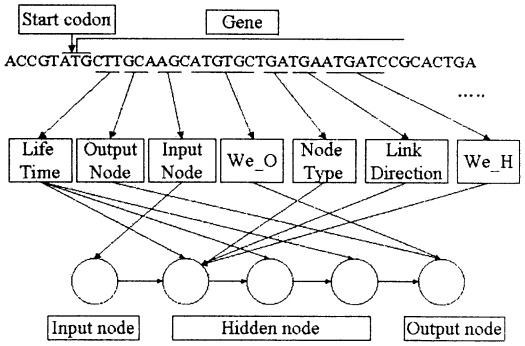


Figure 5. Structure of Gene And Corresponding element

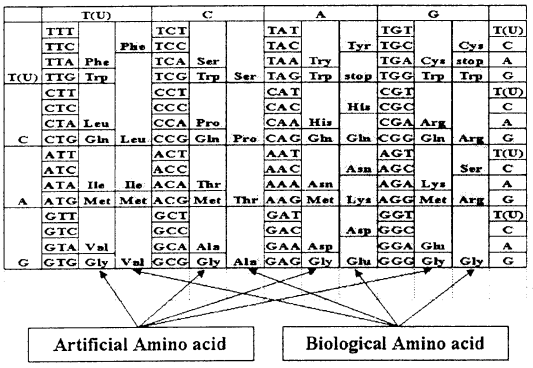


Figure 6. Amino acid table

< **Codon Translation Table** >

	Phe	Ser	Try	Cys	Trp	Leu	Pro	His	Arg	Gln	Ile	Tjr	Asn	Lys	Met	Val	Ala	Asp	Glu	Gly
Life	0			1						2					3			4		5
Dir	Negative (-)									Positive (+)										
Type	Q type							N type					A type							
Out	Output node I									Output node II										
In	Input node I							Input node II					Input node III							

WE Translation Table

WE Translation Table						
	a	b	c	d	e	f
T(U)	-3	0	0	0	0	0
C	-1	1	1	1	1	1
A	1	2	2	2	2	2
G	3	3	3	3	3	3

$$WE = a*(b+c*4^{-1}+d*4^{-2}+e*4^{-3}+f*4^{-4})$$

Figure 7. Amino acid Translation table

4. Evolution of Neural Network using DNA Coding Method

A neural network used in this paper is generated by the genetic information and the given environment. The latter is considered as the number of inputs and outputs and the former consists of the type of node, the WE value and direction of link. A chromosome includes several genes that generate paths as much as the number of genes. Constructing the neural network using DNA coding follows two steps as Fig. 4.

4.1 First Translation Step

First translation consists in a neural network's path and determines characteristic of hidden node, using genetic information of the chromosome. The structure of gene is shown in Fig. 5. Fig. 8 shows the flow of translation from the gene to neural network element, and Fig. 7 shows correspondence between amino acid and parameters defined in this paper. The amino acid table is represented in Fig.6

Unlike creatures, we obtain the same amino acid with respect to similar codon by equalizing a probability of amino acid generation. After finishing the first step to all genes, the second translation step continues.

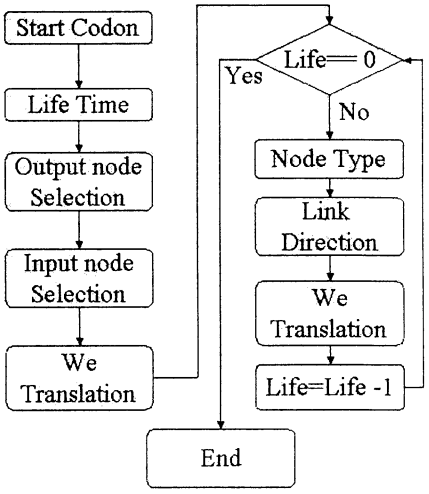


Figure 8. The Flow of Translation from Gene to path of neural network

4.2 Second translation step

In the second translation step, we determine a interconnection of hidden nodes, weights and bias with respect to consistent environment; path , node type, and link direction. A hidden node has one input at the own path, and the other input at another path. A hidden node has weight and bias according to its type like Fig. 1

5. Simulation

In this paper, we applied this method to autonomous mobile robot(AMR) control problem. We assume that AMR has three sensors at the front, left side, and right side. Sensors measure distance from AMR to obstacles. A Sensor`s output is neural network`s input, and A neural network`s output is motor`s input of AMR.

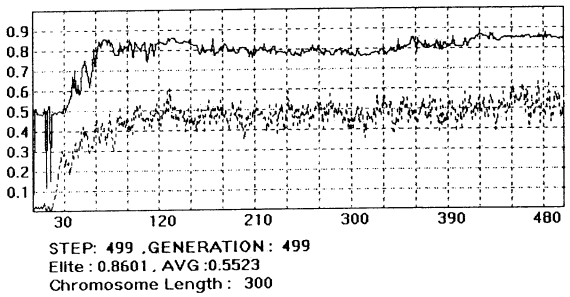


Figure 9. Best Fitness and Average Fitness

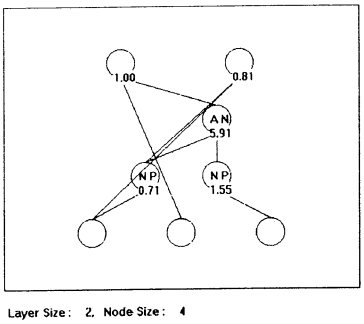


Figure 10. A Neural Network constructed after 500 generation

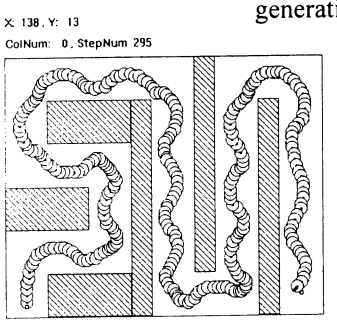


Figure 11. Trace of Best Individual on the Evolution Environment

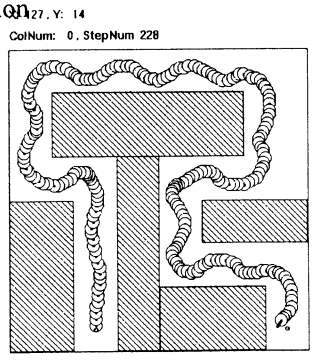


Figure 12. Trace on A Unknown Environment

Fig. 9 shows best/average fitness. Fig. 11 shows the trace of the best individual of 500 generation evolution. In simulation, we set the parameters as follows:

- Population size : 50
- Length of initial chromosome : 300
- Crossover rate : 0.7
- Mutation rate : 0.01
- Number of input node : 3
- Number of output node : 2

6. Conclusion

We proposed neural network evolving method based on DNA Coding Method and verified it by computer simulation. The generated neural network is simple and evolution time is short because we didn`t use learning step during evolving process. We will apply this method to real autonomous mobile robot

Reference

[1] J. A. Freeman, D. M. Skapura, " Neural Networks : Algorithms, Applications, and Programming Techniques" Addison Wesley.

[2] D. E. Goldberg, "Genetic Algorithms in Search, Optimization & machine Learning" Addison Wesley.

[3] C. Jocob and J. Rehder. "Evolution of neural net architectures by a hierarchical grammar-based genetic system", Proc. of the International Joint Conference on Neural Networks and Genetic Algorithms 1993.

[4] F. Gruau, "Neural Network Synthesis using Cellular Encoding and the Genetic Algorithm", PhD thesis at Ecole Normale Superieure de Lyon 1994

[5] T. Yoshikawa, T. Furuhashi, Y. Uchidawa , "Acquisition of Fuzzy Rules of Constructing Intelligent Systems using Genetic Algorithm based on DNA Coding Method" Proceedings of International Joint Conference of CFSA/IFIS/SOFT`95 on Fuzzy Theory and Applications

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Application of the Neural Networks for the Diagnosis of Blood Constituents

Mieko Tanaka-Yamawaki

Department of Computer Science and Systems Engineering, Miyazaki University, Miyazaki 889-2192, Japan
mieko@cs.miyazaki-u.ac.jp

and

Satoshi Yoshida

Research Laboratory Center, Oita Medical University, Hasama-cho, Oita 879-5593, Japan

Keywords : neural network, noninvasive diagnosis, health-care product, FTIR

Abstract

We apply the neural network architecture in constructing an on-line, real-time diagnosis program in PC that is connected to an infrared Fourier spectra analyzer, in order to check the health-condition in the non-invasive manner. Since the best result is expected when the connection weights in the neural network are personally trained for individual users, we suggest implementing this system in a home robot system together with other functions such as a mental condition analyzer as well.

1. Introduction

As we are deeply concerned to establish a health care system that can be managed by individuals not overly dependent on the social welfare system, it is important to have a handy, noninvasive diagnosis equipment to be placed at every home. We propose here a model for such a system composed of a personal computer equipped with a real-time diagnosis program, that is linked to a measurement system of Fourier transform infrared spectral analysis (FTIR), which one of the authors has constructed recently for the purpose of measuring the constituents of blood inside mucous membrane of the human mouth in the non-invasive and non-destructive way. This system outputs the Fourier transform of the infrared spectra, by analyzing the blood contents from the reflected infrared ray obtained simply by touching a rod-shaped apparatus on the surface of the human oral mucous membrane for one minute [1].

The program is designed to identify the characteristics of the data instantly upon receiving the FTIR spectra from the measurement system in the laboratory. We prepare the neural network to separate various conditions of the same subject in advance, by taking data and analyze them by statistical methods as well as the chemical tests as usually done in the laboratories. Once the network learns various features of this specific subject, this personalized system can work for the home-diagnosis system for geriatric diseases such as diabetes or hyperlipidemia. It will also work for warning the obesity, which is often found as a preliminary signal of those diseases.

2. Neural Network Program

The program is set up as follows. The input layer consists of N units, corresponding to the spectral strengths for N typical wave numbers. The output layer consists of M digital (0 or 1) units, corresponding to the M types of the blood conditions. The connecting weights between all the possible pairs of N input units and M output units are trained together with the threshold values of each unit. Currently we take the values of (N, M) to be (16, 5).

We have trained the connection weights by using the sample data classified by means of statistical analysis into M different types extending twelve different degrees in strength. In the minimal model of the single-layered neural network, i.e. having no hidden layer units, the final weights have been obtained when the iteration converges after several hundred turns.

By using the connection weight obtained in the method stated above, we have studied the performance of the program by using a large number of data sets taken by the same individual in various time and conditions.

3. Data Analysis

Based on the characteristics of the sample data used to train the connection weights, we attempted to separate the data into five patterns:

Type 1: Fatty-acid-like
Type 2: Glucose-like
Type 3: Lipid with ester-bond
Type 4: Protein-like
Type 5. Glucose-like

2921.7	0.00110354	***				
2852.2	0.00103174	***				
1741.1	0.0007273	***				
1677.5	0.00056097					*
1648.5	0.00472245					*
1546.3	0.00273335				*	
1515.4	0.00109288	***		*		
1455.6	0.00152956		**		*	
1417.1	0.0005628		**		*	
1395.8	0.00057522		**	*	*	
1366.7	0.00008566		**	*		
1314.8	0.00040479				*	
1241.5	0.00040826				*	
1123.9	0.00024069			*		
1079.5	0.00059022		**			
1023.6	0.00005661	***				
Wave number	Data intensity	1	2	3	4	5

Table 1. Standard data from which training data are generated. Asterisks indicate the locations of wave numbers responsible for the five different types.

4. Concluding Remarks

So far it has been possible to classify the patterns of test data by using the weights obtained in the above-mentioned method in the practical degree of accuracy.

This system may be considered to be installed in a home robot system equipped with various functions as a home doctor together with the human random generation testing system which examines the alertness level of the brain, as a home-diagnosis system to warn Alzheimer's disease in the non-invasive way [2].

References

1. Yoshida, S., *Obesity Research*, vol.3, S761-S767 (1995).
2. Tanaka-Yamawaki, M., Masuda, H., and Kawagoe, M., *Proceedings of 3rd International Workshop on Robotics and Artificial Life*, 610-613 (1998); Tanaka-Yamawaki, M., *Proceedings of 5th International Conference on Neural Information Processing*, 215-218 (1998).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Recognition of the sequential motion of the hand by the GRBF network

Miwako Hirakawa
Fukuoka International University
Gojo,Dazaifu-city,Fukuoka 818-0193,Japan
e-mail: miwako@fukuoka-int-u.ac.jp

Masami Ishibashi, Taeko Murakami,
and Masahiro Okamoto
Kyushu Institute of Technology
Iizuka-city,Fukuoka 820-8502,Japan

Abstract

Poggio and Edelman have proposed a method of automatic acquistition of 3D model from 2D views rotating 3D object and have named GRBF(Generalized Radial Basis Function) network. They have shown that for each object there exists a smooth mapping from an arbitrary view to its standard view and that the such a mapping can be learnt from a sparse data set. In this paper, we have applied this network to the recognition of sequential motion and designed the system to capture motions of the hand with the GRBF by using data glove. Since sequential motion consists of a lot of frames, we can pick up typical several canonical frames of the motion and these frames can be applied to the GRBF network for learning. The trained GRBF network could achieve a recognition over 84%.

Keyword : neural network, 3D-object recognition, structural learning, Gaussian GRBF, hand-shape change, hand's motion capture

1 Introduction

It is usually a very difficult problem to acquisition of 3D models from images automatically. One way to overcome the need of 3D model is to exploit methods for representing objects by a collection of 2D views (2D projection) rotating 3D object. Poggio and Edelman [1], Edelman and Poggio [2], Poggio [3], Poggio and Girosi [4], and Maruyama et al. [5] have proposed such a view-based object-recognition method named GRBF(Generalized Radial Basis Function) network which relys on multiple 2D views instead of 3D models. This network resembles to the conventional artificial neural network, however, each unit in a hidden layer is represented by radial basis function such as Gaussian. In this paper, we have applied GRBF network to the recognition of hand shape-changes such as grasp, flap and open. We show their performance by computer simulations and by using data glove.

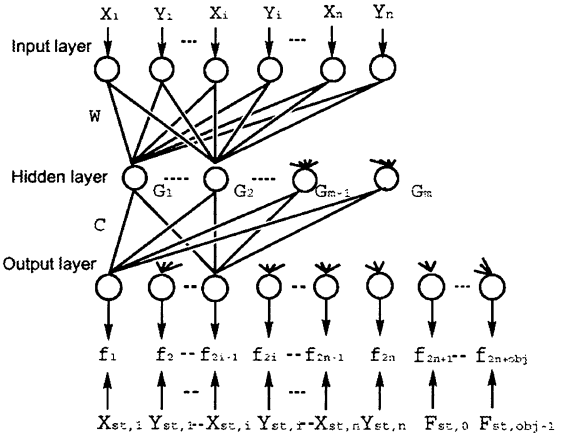


Figure 1: GRBF network structure.

2 GRBF network for rigid objects

A Gaussian GRBF network is shown in Figure 1, which consists of three layers (input layer, one hidden layer, output layer). Since 3D object is supposed to be a collection of 2D coordinates (perspective view) of the feature points on the image, each 'snap-shot' 2D view is represented as a 2n vector *X1, Y1, X2, Y2, ..., Xn, Yn* of the coordinates on the image plane of n labeled and visible feature points on the object. We assume here that all features are visible. Each basis unit in hidden layer can be represented by Eq.(1), where *Tx* and *Ty* are centers of Gaussian functions for *X*-axis and *Y*-axis, respectively, *m* is the number of basis units in hidden layer. The *W* is supposed to be binding coefficient (weight) between units in input layer and those in hidden layer. The resulting value *Gk* ($k = 1, 2, .., m$) can be regarded as the 'activity' of the unit. The output of GRBF network is the linear superposition of the activities of all the basis unit *Gk* in hidden layer as shown in Eq.(2). Since GRBF network is designed to explore the mapping that transforms each 'snap-shot' 2D view to its standard 2D view of the object, the

objective function of the network can be written in Eq.(3), where (Xst,i, Yst,i) (i = 1, 2, ..., n) is standard coordinate of the i th object, N is the number of learning data set (examples). In order to learn several kinds of 3D object in one Gaussian GRBF network, teacher flags for recognition of object, Fst,i (i = 0, 1, ..., obj-1), were prepared, where obj represents the number of 3D objects to be learnt. For example, suppose three kinds of objects should be learnt, when 2D projection view of the 1st. object is given to the system, ($Fst,0$, $Fst,1$, $Fst,2$) is set to be (1,0,0). For the case of that 2D projection view of the 2nd and 3rd objects are given , ($Fst,0$, $Fst,1$, $Fst,2$) is will be (0,1,0) and (0,0,1), respectively. In the same manner as in Eq.(2), flag output (f_{2n+p+1}, (p = 0, 1, .., obj-1)) will be calculated according to Eq.(4). In this study, we also consider the mean squared error between flag output and teacher flag to define the objective function (H) as in Eq.(5), where A is an arbitrary coefficient.

$$G_k = exp\left(\sum_{i=1}^{n} -(X_i - T_{xk})^2 - W_{k(2i-1)}^2\right) \times exp\left(\sum_{i=1}^{n} -(Y_i - T_{yk})^2 - W_{k(2i)}^2\right) \quad (k = 1, 2, .., m) \quad (1)$$

$$f_j = \sum_{k=1}^{m} C_{k,j} G_k \quad (j = 1, 2, ..., 2n) \quad (2)$$

$$E = \sum_{q=1}^{N}\sum_{i=1}^{n}\left((X_{st,i} - f_{2i-1})^2 + (Y_{st,i} - f_{2i})^2\right) \quad (3)$$

$$f_{2n+p+1} = \sum_{k=1}^{m} C_{k,2n+p+1} G_k \quad (p = 0, 1, .., obj\text{-}1) \quad (4)$$

$$H = \sum_{q=1}^{N}\left(\sum_{i=1}^{n}\left((X_{st,i} - f_{2i-1})^2 + (Y_{st,i} - f_{2i})^2\right) + A\sum_{p=0}^{obj-1}(F_{st,p} - f_{2n+p+1})^2\right) \quad (5)$$

To use this network first we have to decide on the number of basis units (m) is the hidden layer. Like most other feedforward network, usually we do not have any guidelines for this and hence learning with trial and error becomes inevitable. We have developed structural learning algorithm which is focused on automatic adjustment of the number of basis units in hidden layer [6].

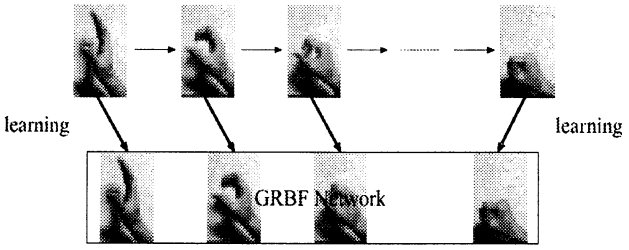


Figure 2: Concept of GRBF network for hand's motion capture.

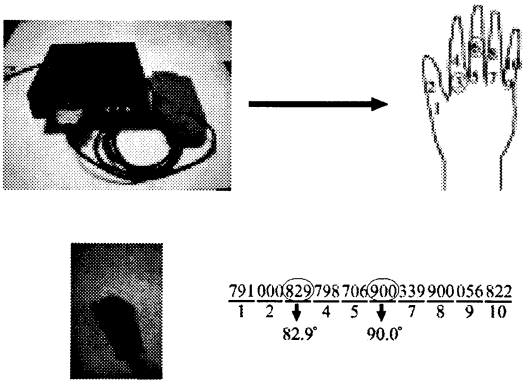


Figure 3: Data globe and joint's position of hand.

3 GRBF network for hand shape-change

Next, we have applied GRBF network to the recognition of hand shape-changes. Figure 2 shows the sequential motion of grasp. Since sequential motion consists of a lot of frames, we can pick up typical several canonical frames of the motion and those frames can be applied to the GRBF network for learning (see Figure 2). In this study, we used data glove as in Figure 3 to collect the angle information of hand's joints. After the calibration, this data glove can simultaneously calculate the angle (between 0° and 90°) at 10 joints of one hand at an arbitrary frame (see Figure 3). These data are fed to the GRBF system shown in Figure 1, where n=5 and $X1$, $Y1$, $X2$, $Y2$, ..., $X5$, $Y5$ represent angle information of 10 joints, respectively. As an example $Y1$ and $Y5$ show the angle data at the joints 2 and 10 in Figure 3, respectively. At a certain frame of hand motion shown in Figure 3, the angle data at joints 3 and 6 are 82.9 ° and 90.0° , respectively.

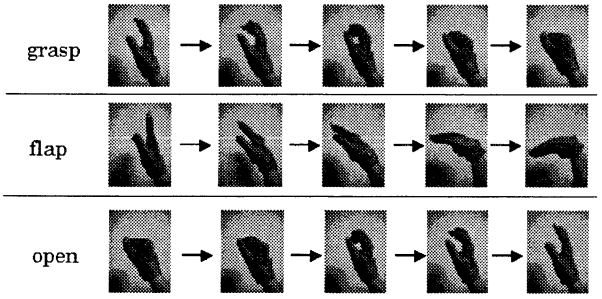


Figure 4: Three kinds of hand-change motions.

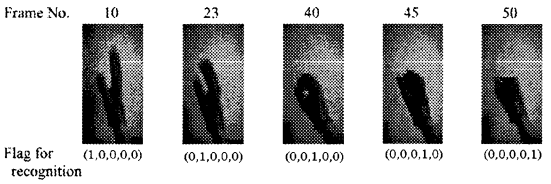


Figure 5: Teacher frames of 'grasp' and their teacher flags for recognition.

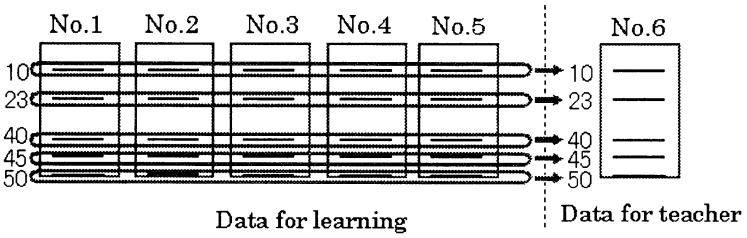


Figure 6: Training procedure of hand-change motion

4 Recognition of hand shape-change

Figure 4 shows three kinds of hand-change motions; grasp, flap and open. Suppose one motion, such as grasp, consists of sequential 50 frames and 5 canonical frames (in this study, frame number, 10, 23, 40, 45, 50 (see Figure 5)) are picked up among them for learning. We set up 5 kinds of flag for recognition (teacher flag, Fst,i (i = 0, 1, ..., 4)) as shown in Figure 5. The flag number (1,0,0,0,0) is assigned to the frame number 10, (0,1,0,0,0) is to the frame number 23, (0,0,1,0,0) is to the frame 40, (0,0,0,1,0) is to the frame 45, (0,0,0,0,1) is to the frame 50. As shown in Figure 6 preparing 6 sets of angle data of 10 joints at frame 1 to 50 and picking up the data at the frame number 10, 23, 40, 45 and 50 from them, 5 sets of angle data were given to GRBF network as training data and the remaining 1 set of angle data was used for standard angle of this motion (teacher angle). At learning process, for instance, when the training angle data at the frame 10 (No. 1 – 5 at column 10 in Figure 6) are given to the GRBF network, Xst,i, Yst,i in Eq.(5) indicate angle data at the frame 10 of teacher (No.6 at column 10 in Figure 6) and teacher flag, Fst,i (i = 0, 1, ..., 4), is assigned to (1,0,0,0,0), respectively. In the same manner as this, given the training angle data sets of five canonical frames (10, 23, 40, 45 and 50 frame), the GRBF network explore and tune the parameters (Tst, Tst, W, C) in order for the objective function H in Eq.(5) to take the value less than a certain threshold. The optimization algorithm has been described elsewhere [6]. After the learning, fixed the estimated parameters in GRBF network, known or unknown sequential 50 angle data sets were applied frame by frame to the final trained GRBF network, the order of the values of flag output was evaluated. If the order of flag output ($f11$, $f12$, $f13$, $f14$, $f15$) continuously shifts (1,0,0,0,0)→(0,1,0,0,0)→(0,0,1,0,0)→(0,0,0,1,0)→(0,0,0,0,1) with the frame, this motion can be regarded as the template hand motion. In this case, since the GRBF network was trained to generate the corresponding teacher flag by using training angle data at the frame 10, 23, 40, 45, 50, the network should provide the corresponding output flag when the angle of those frames are given; output flag ($f11$, $f12$, $f13$, $f14$, $f15$) should be (1,0,0,0,0) for the case of the frame 10, (0,1,0,0,0) for the frame 23, (0,0,1,0,0) for the frame 40, (0,0,0,1,0) for the frame 45, and (0,0,0,0,1) for the frame 50. Providing angle data at other frames, however, what value of flag output will the network generate? This corresponds to the interpolation capability of the trained GRBF network. As shown in Figure 7, when the angle data at the neighboring frame on the trained one (10, 23, 40, 45 or 50) are given to the trained GRBF network, it is highly expected that output flag ($f11$, $f12$,

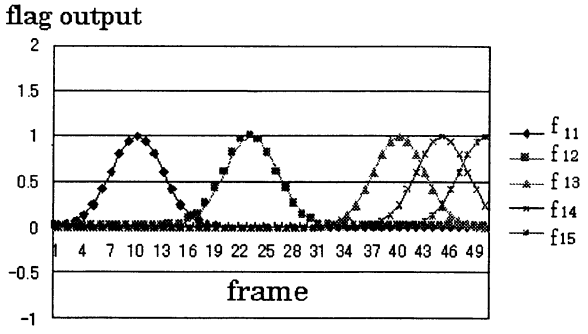


Figure 7: Expected shift of flag output.

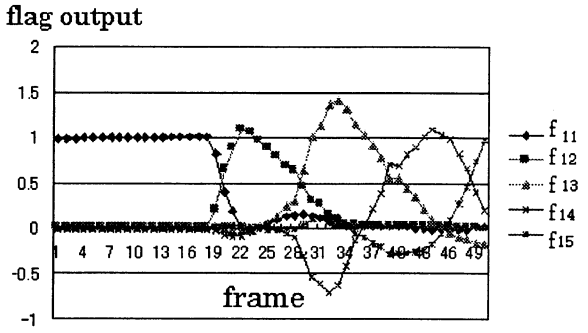


Figure 8: Shift of flag output in the case where the training data set of 'grasp' are given.

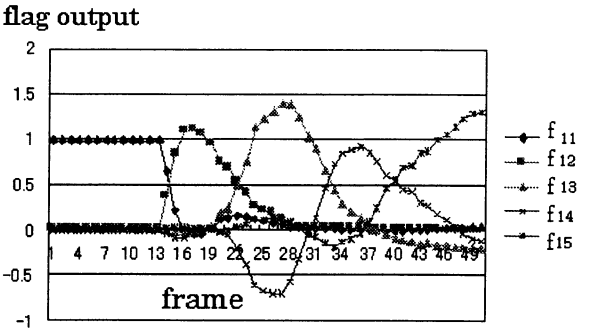


Figure 9: Shift of flag output in the case where unknown data set of 'grasp' are given to the GRBF network trained 'grasp'.

f13, f14, f15) will come near to the values of the corresponding teacher flag and that the nearer frame is given, the value of output flag is closer to that of teacher flag. Figure 8 represents the shift of flag output in the case where the training data set (No. 1 to 5 in Figure 6) of 'grasp' are given to the trained GRBF network. From frame 1 to 18, flag output (*f11, f12, f13, f14, f15*) takes the value (1,0,0,0,0). The curves of flag output *f12, f13, f14* and *f15* have the peak at the frame 23, 34, 45, 50, respectively, profiles of which are similar to the expected ones (Figure 7) except for *f13*.

In order to evaluate and to recognize sequential motion of the hand only by the values of flag output (*f11, f12, f13, f14, f15*), The following recognition rule was assigned to each flag output:

(1) At each frame, flag output more than 0.5 is only adopted. The maximum value of flag output is 1.0.

(2) When more than two flag output is adopted by the rule (1), flag output the value of which is the biggest is adopted.

(3) When the values of all flag output at a certain frame are less than 0.5, no flag output is adopted. At that frame; (*f11, f12, f13, f14, f15*) = (0,0,0,0,0).

(4) If the order of flag output (f11, f12, f13, f14, f15) continuously shifts (1,0,0,0,0) → (0,1,0,0,0) → (0,0,1,0,0)→(0,0,0,1,0)→(0,0,0,0,1) with the frame, this motion can be regarded as the template hand motion (in this study, 'grasp').

(5) At the rule (4), (0,0,0,0,0) is inserted at a certain frame, this motion can also be regarded as the template hand motion.

Applied these recognition rule to the profile shown in Figure 8, the order of flag output (*f11, f12, f13, f14, f15*) has continuously shifted (1,0,0,0,0) → (0,1,0,0,0) → (0,0,1,0,0) → (0,0,0,1,0) → (0,0,0,0,1) with the frame. Next, providing 100 sets of unknown (not used for training or learning) 50 sequential angle data of 'grasp' frame by frame to the GRBF network trained 'grasp', the correct recognition score was examined by evaluating the order of the values of flag output. Figure 9 represents the profile of flag output with the frame in the case where one set of unknown 50 sequential angle data are given to the trained network; the profile of Figure 9 is very similar to that of Figure 8. The trained GRBF network could achieve a correct recognition score over 84% among 100 unknown data sets of 'grasp'.

How will the profile of flag output be going when the angle data of other hand motion are provided to the GRBF network trained 'grasp'? Figure 10 shows the profile of flag output when the angle data of 'flap' are given; the resulting flag output with the frame is (1,0,0,0,0), which means that the hand motion of 'flap' is the same as that of frame 10 in 'grasp'. (see Figure 5). The trained GRBF network could perfectly recognize the difference in the motion of hand between 'grasp' and 'flap'. We also have a remarkable point; as

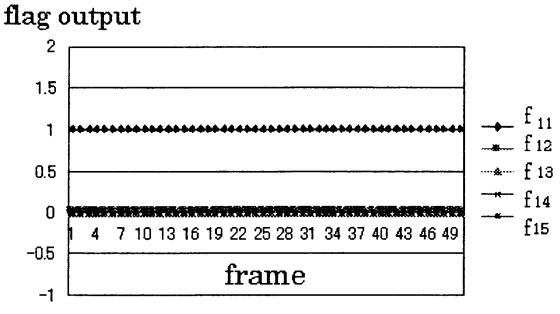


Figure 10: Shift of flag output in the case where unknown data set of 'flap' are given to the GRBF network trained 'grasp'.

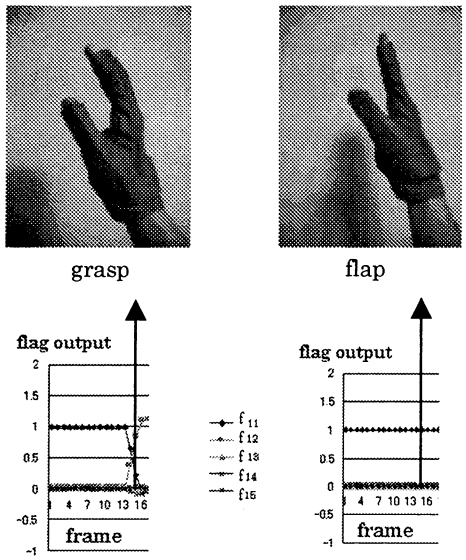


Figure 11: Snap shots of the frame 15 of 'grasp' and 'flap'.

shown in Figures 9 and 10, the flag output (1,0,0,0,0) shifts to (0,1,0,0,0) at frame 15 in Figure 9, while the flag output (1,0,0,0) keeps as it is after frame 15 in Figure 10. This result means that the GRBF network could recognize the difference in the motion of hand between 'grasp' and 'flap' at the stage of frame 15 (early stage of hand-motion change). Figure 11 represents the snap-shots of 'grasp' and 'flap' at the frame 15. The GRBF network can recognize this slight difference in motion between two frames. Figure 12 represents the simulation result when the angle data of 'open' (see Figure 4) are given. Applied the recognition rules (1) - (5) to Figure 12, the order of flag output (f_{11}, f_{12}, f_{13}, f_{14}, f_{15}) has continuously shifted (0,0,0,0,1) → (0,0,0,1,0) → (0,0,1,0,0) → (0,1,0,0,0) → (1,0,0,0,0) with the frame, which is the inverse order of flag output to that in Figure 8; 'open' is the inverse motion of hand to 'grasp'.

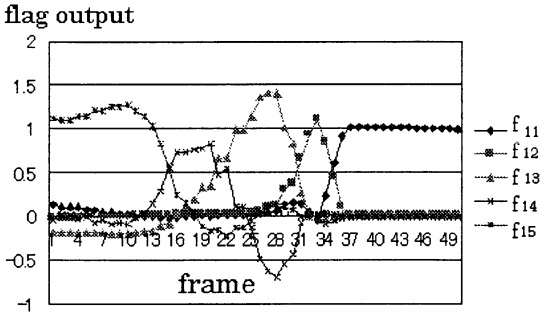


Figure 12: Shift of flag output in the case where unknown data set of 'open' are given to the GRBF network trained 'grasp'.

5 Conclusions

(1) We have designed the GRBF network for the recognition of the sequential motion of the hand.

(2) The trained GRBF network showed a high correct recognition score.

(3) It is highly expected that the trained GRBF network can recognize the difference in hand-motion before the whole motion change is not completed.

References

[1] Poggio T. and Edelman S., (1990), "A network that learns to recognize 3D objects", Nature, 343, pp.263-266.

[2] Edelman S. and Poggio T., (1990), "Bringing the grandmother back into the picture: a memorybased view of object recognition", MIT AI Lab. Memo, No.1181.

[3] Poggio T., (1990), "A theory of how the brain might work", MIT AI Lab. Memo, No.1253.

[4] Poggio T. and Girosi, F., (1990), "Regularization algorithms for learning that are equivalent to multilayer networks", Science, 247, pp.978-982.

[5] Maruyama, M., Teraoka, T., Abe, S., (1994), "Recognition of 3D flexible objects by GRBF", Biological Cybernetics, 70, pp.377-385.

[6] Kinoshita, K., Katsuki, T., Miyazaki, T., Hirakawa, M., Okamoto, M., (1998), "Recognition of shape and shape-changes in 3D objects by GRBF network", Proc. of 3rd. Int. Symp. on Artificial Life and Robotics (AROB III'98), 1, pp.228-233.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Forecasting and precision on using multi-layer neural networks: *Introducing of auxiliary functions*

Tomoki TESHIMA, Hanxi ZHU, Tomoo AOYAMA , and Ikuo YOSHIHARA
The Faculty of Engineering, Miyazaki University
Gakuen kibanadai-nishi 1-1, Miyazaki 889-2192, Japan
e-mail: t0b217u@cc.miyazaki-u.ac.jp

Keywords: recurrence, forecasting, and time-series data
Abstract:

There is a method to forecast time-series phenomena, which is based on functions of multi-layer neural networks, and a recurrent representation of functions. The method is practical, but has some insufficient points. In order to remove a defect, we introduce a method using auxiliary functions. The method embosses a character in phenomena. We believe that the embossment is necessary to forecast accurately a future of the phenomena. However, such embossment has many faults. Therefore, we evaluated the effects of the auxiliary functions. In the evaluation, we used a rectangular function that is the most important in area of electronics. The evaluations showed remarkable precision improvement for forecasting.

1. Introduction

We discussed a general method to forecast time-series phenomena. The method was based on functions of multi layer neural networks, and a recurrent representation of functions [1-3].

There are many kinds of property in time-series phenomena. When we predict a future of the phenomena, we use the property involuntarily. We believe that the property can be represented by functions. We try to forecast phenomena by using neural networks and the functions. In the trials, there are assumptions. One is that information to forecast the futures of phenomena is distributed homogeneously in observed data. Another is that phenomena have a recurrent part.

We already discussed long-range forecasting for chaos in a previous paper [ ]. The chaos was deterministic, and could be evaluated by numerical means strictly. In long-range forecasting, the recurrent representation of functions for multi descriptors was effective. On the other hand, natural phenomena may be not deterministic. Even if they are determined, the direct equations are too complex to decipher. But, we use neural networks; because we believe that they have a facility to reproduce complex phenomena. We find probability of forecasting in the facility. If the facility were not enough, as an example, we introduce a periodic property in the phenomena as a periodic function. The function can be used with plural kinds. We call them auxiliary functions. The phenomena are expressed by an observation and plural auxiliary functions. That is an assumption; and we couldn't find reference about the assumption. There are some problems; an example is following. Although we must forecast a future of phenomena on use of observed data, the assumption breaks the principle. In the paper, we discuss it.

2. Two conceptions for prediction

When we forecast phenomena by using neural networks, at first we construct a neural network by learning equations. The forecasting is evaluated based on the constructed network; at the time, there are two choices for adopting prediction data.

One method uses last observed data just before the forecasting. The conception is called as "one step ahead prediction" or "short-range forecasting". It is similar to Takens' embedding theory. But, in the prediction principle, a basic quantity is not corresponded with local information such as the Jacobian. Moreover, the quantity is not gotten by the last data, but information from rather past data. Then, it is somewhat extension for Takens'.

Second method uses output data calculated only by the neural network. The conception is called as "long-range forecasting". It is out-of-range of Takens' embedding theory, whose efficiency is only examined by numerical calculations. It is natural that the forecasting from the former is higher precision than that of later; however, we believe the second method more useful in practical.

We are sure that the future of phenomena must be forecasted by the second method. But, must the auxiliary functions, which were introduced for emphasizing some property, be predicted by the second? We believe that they may be done by the first method. They can be defined beforehand. We can adopt the first method.

3. Precision for recurrent representation

In neural networks, input data are multi dimensional. In order to simplify following discussion, we put the multi dimensional data on a multi dimensional space. From now on, input data are points on the space, so we write learning data as learning points.

The recurrent representation of functions is a trace on the multi dimensional plane, whose dimension number is equal to the window width plus one. The trace is a closed curve in case of periodic functions. It is desirable that the curve is uniform at

any point for forecasting, and the sampling points are distributed on the curve homogeneously. If the curve isn't closed, the original function is not periodic. Even if the original is periodic, in case of insufficient sampling (or observations), the curve isn't closed. On such a case, the sampling points are an over concentration as following figure 1. Forecasting may not be done as for the lack part. However, outputs from neural networks don't vary so rapidly under the characters of monotonously increasing of sigmoid function. The movements of neural networks compensate the lack part. This is a reason why neural networks predict a future in case of insufficient observations.

4. Auxiliary functions

If a phenomenon were periodic, their sampling data would have same property. If we add another periodic sampling to them, the property of whole data is invariable. Another data may emboss the periodicity of phenomena. If we find a character in a target phenomenon, and the character is hidden because of adopting a wide window width, we would allow embossments of the unremarkable character. We call the supplement data auxiliary data, and name the origin of the data auxiliary function. Plural kinds of the functions can be adopted in response to characters of the target. The affection of the auxiliary functions is listed in figure 1.

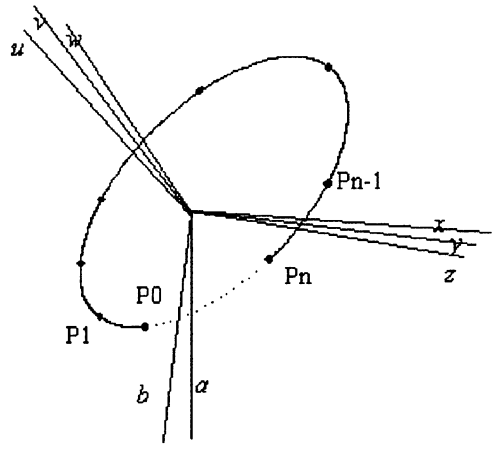


Figure 1. *Sampling points "Pn" on a multi dimensional space that has u, v, , and b coordinate axes*

If a phenomenon were periodic, the trace of insufficient sampling is not closed. In the case, if auxiliary functions were introduced, the lack part of the closed curve would be supplemented as a dotted curve in the figure.

Since we aim the long-range forecasting, we don't adopt one-step prediction like as Takens' embedding theory. The course is carried out for phenomena, but isn't done for auxiliary data. The auxiliary data are not forecasting targets; and they are guides to point out characters in the phenomena. As for them, one-step prediction can be adopted.

5. Numerical calculations

Uncertain hypothesis must be examined in practical cases. We test actions of auxiliary functions in the section. First is an effect, and second is a problem. We used a rectangular function as examination target, whose expression was sin(x)+sin(3x)/3+sin(5x)/5.

5.1 Effects of rectangular function

We adopted a rectangular function that is f(x)= sin(x)+sin(3x)/3+sin(5x)/5, 0<x<2pi. The number of sampling is 40(=n) for the learning. The data set has periodic property. But, adopting a window width for calculations of neural networks, the property is lost. The ratio is (n-w)/n, where "w" is a window width. Under the condition, using ANN [4] the forecasting results are listed in figure 2 and 3.

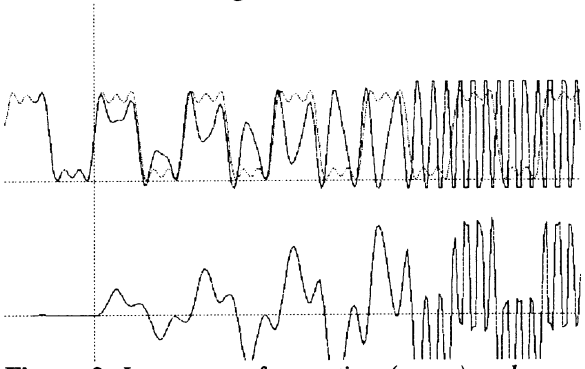


Figure 2. *Long-range forecasting (upper) and error (lower) for y=sin(x)+sin(3x)/3+sin(5x)/5; The rectangular function is scaled within [0,1].*

The left side of a dotted vertical line means a learning term (40 points, window width is 12). The right side is forecasting. effective number of the learning data is 28, and forecasting points are 280. In the first graph, the solid curve is results from ANN-neural-network, and the plotted curve is true values of the rectangular function. Output values from ANN are cut off [-0.1,1.1].

In the second graph, the solid curve is differences between forecasting values and true ones. The case is forecasting when there is 70% information in the learning term.

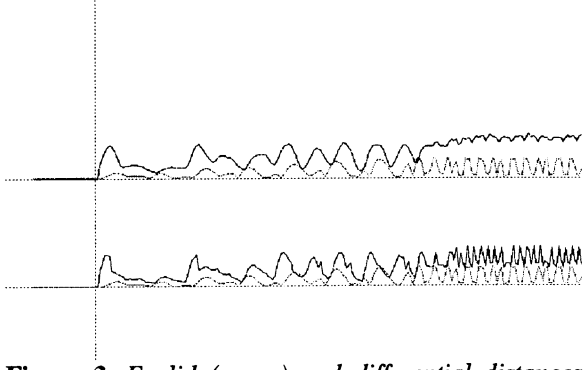


Figure 3. *Euclid (upper) and differential distances (lower): In the graphs, the plotted lines are square of the differences. The solid curves are Euclid and differential distances.*

The figure 2 and 3 show that long-range forecasting is impossible for long-terms. The reason would be that lack of periodicity could not be compensated. Neural networks have the compensation facility certainly, however their facility are not sufficient in case of the figure 2.

Embossing hidden periodic property in learning data, we introduced two auxiliary function that were cos(x) and cos(3x). The auxiliary functions could be sin(x) and sin(3x), but they were components of the original rectangular function, therefore they wouldn't be appropriate ones. The required property for the auxiliary function is not components of target, but property. Since cos(x) has same periodic property to sin(x), it can be adopted. Cos(3x) was necessary to forecast short-cycle of the target. If it were not, the forecasting would be done without the short-cycle. The calculation results with the auxiliary functions are listed in figure 4, 5, and 6.

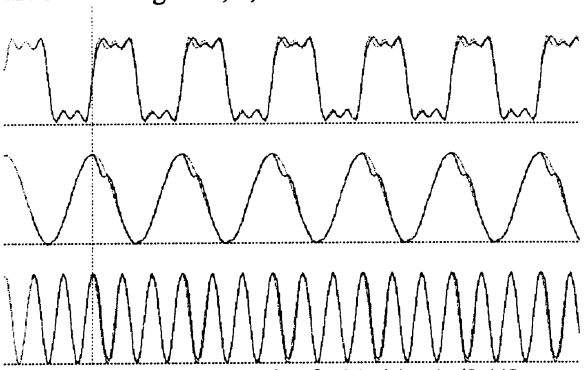


Figure 4. *Forecasting for f={sin(x)+sin(3x)/3 +sin(5x)/5, cos(y), cos(3z)}: As for x-part, long-range forecasting is applied. For other-parts that are auxiliary parts, short-range is applied. The all functions are scaled in [0,1].*

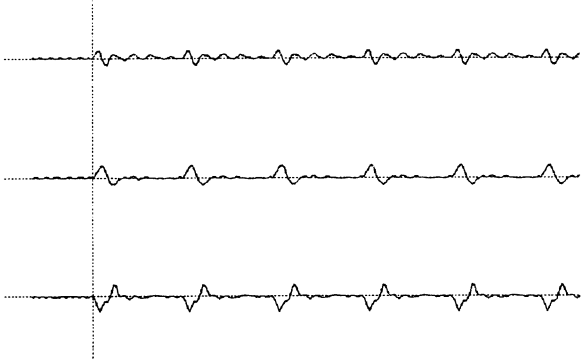


Figure 5. *Differences between forecasting values and true ones: From upper to bottom, graphs are for x-, y-, and z-parts.*

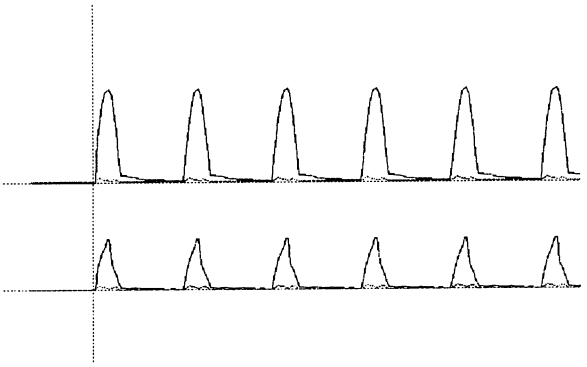


Figure 6. *Euclid (upper) and differential distances (lower):In the graphs, the plotted lines are square of the differences. The solid curves are Euclid and differential distances. Scales of the figures are same as figure 4.*

The effects of the auxiliary functions are remarkable in the figure 4. The long-range forecasting for rectangular function is almost complete. The short-range forecasting for auxiliary functions is swayed slightly at the lack-part for periodic conditions. The swaying parts are corresponded to peaks of Euclid or differential distances. On the parts, reliability for forecasting is rather lower. This is a effect for introducing the auxiliary functions.

Appropriate auxiliary functions are adopted in the case. If we understood whole characters of the targets, such appropriate introductions would be enabled. However, this is impossible in actual case. We should investigate in case of adopting inappropriate auxiliary function.

5.2 Problems of auxiliary function

Even if we may find a major component in phenomena, we can't do minors correctly. That is very difficult, then, the frequency of a minor-cycle in the target wouldn't be determined. When we introduce an auxiliary function for the minor component, an inappropriate function might be used. Such a situation would be always occurred. Therefore, we examined hazards of inappropriate auxiliary functions, that is two auxiliary function, cos(x) and cos(2x). The later function cos(2x) is inappropriate. The calculated results are listed in figure 7, 8, and 9.

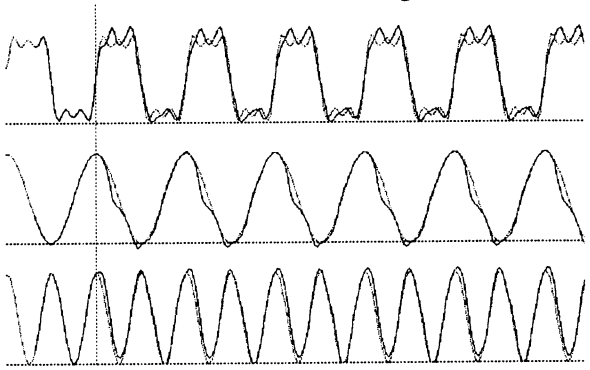


Figure 7. *A hazard in forecasting for f={sin(x) +sin(3x)/3+sin(5x)/5, cos(y), cos(2z)}: Cos(y) and cos(2z) are used as auxiliary functions. The later function has inappropriate frequency. All functions are scaled in [0,1].*

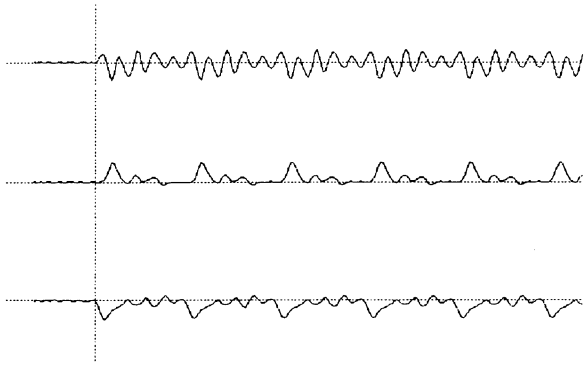


Figure 8. *Differences between forecasting values and true ones: From upper to bottom, graphs are for x-, y-, and z-parts.*

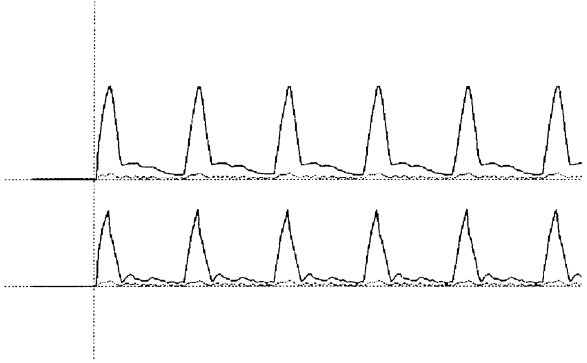


Figure 9. *Euclid (upper) and differential distances (lower):In the graphs, the plotted lines are square of the differences. The solid curves are Euclid and differential distances. Scales for the figures are same as figure 7.*

We can't find that the auxiliary function is appropriate or not, on forecasting-curves. But, the reasonable property can be checked by Euclid or differential distance. If one compares figure-6 with figure-9, he would find that slopes between peaks are higher in case of inappropriate auxiliary function. Or, comparing with differences of sways between forecasting-curves of auxiliary functions, one finds that the intensity of the sway is larger on inappropriate function. This is shown in figure 4 and 7.

6. Conclusion

We discussed meanings of the recurrent representations of functions and forecasting by using multi-layer neural networks. We introduced auxiliary functions from the discussion. The function embosses a character in the phenomenon. The embossment is necessary to forecast accurately a future of the phenomena, because the character may be hidden behind decreasing effective number of the data. The decreasing is inevitable to adopt windows for recurrent representations. It might produce an arbitrary result; therefore, the introduction must be strongly limited within the compensation on using the windows. The results from the auxiliary functions must be examined by using Euclid or differential distances, and moreover, they should be checked on sway-curves for short-range forecasting-curves of the auxiliary functions. We evaluated the affection of the auxiliary functions in forecasting of a rectangular function. The evaluations showed remarkable precision improvement for forecasting, and at same time, they instructed the above-mentioned limitations.

References

[1] Tomoo AOYAMA, Yoshimi ISU, and U.NAGASHIMA, "Extrapolations by using neural-networks and a recurrent representations of functions", IPSJ SIG Notes, 95-HPC-59 (1995.12.11), in Japanese.

[2] Hanxi ZHU, Tomoo AOYAMA, and Ikuo YOSHIHARA, "Forecasting and precision on using multi-layer neural network", Proceedings of the 14th KACC, E-218 (1999.10.14).

[3] M. ONODERA, U.NAGASHIMA, H.YOSHIDA, T.AOYAMA, and H.HOSOYA, "Neural Network Reproduction of Time Series Data with Varying Amplitudes and Frequencies", J. Chemical Software, Vol.4, pp.57(1998), in Japanese.

[4] We don't know certainly when "Analogue type Neural Network (ANN)" was used. We adopted it in paper; Tomoo AOYAMA, Yuji SUZUKI, and Hiroshi ICHIKAWA, "Neural Networks to quantitative structure-activity-relationships", J.Med.Chem, 33, 2583 (1990).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Long range forecasting for chaos by using multi-layer neural networks: *Ikeda's chaos*

Hanxi ZHU, Tomoki TESHIMA, Tomoo AOYAMA, and Ikuo YOSHIHARA
The Faculty of Engineering, Miyazaki University
Gakuen Kibanadai-nishi 1-1, Miyazaki 889-2192, Japan
E-mail: t0b217u@cc.miyazaki-u.ac.jp

Keywords: recurrence, forecasting, and chaos
Abstract:

We discuss a general method to forecast movements of the chaos. The method is based on functions of multi-layer neural networks and a recurrent representation of functions. We are sure that forecasting of the chaos is an important problem. Movements of the chaos are extremely complex, and similar to natural phenomena. The forecasting for the chaos has been studied based on the embedding theory. They are called "one-step prediction." The embedding theory is effective and useful, and the results are accurate. But we wish to know long-term futures in practical objects. We need to take one step forward from the embedding theory. Our target is the long-term forecasting.

1. Introduction

Many studies of the extrapolation for time dependent phenomena have been published. In the studies, there were some methods that are adopted neural networks [1,2]. Using neural networks, explicit descriptions (functions) are not necessary. Futures of phenomena are predicted from observed data only. It is very useful and practical, but the usefulness is a defect at same time. If explicit functions are unknown, it is hard to calculate precisions of extrapolations. We often find examples of prediction in published papers [3], but hardly discover examinations using neural networks. We consider that the neural network is one of functions. The function is defined at learning data, and it is discrete essentially. But we use it as continuous, and make predictions. It is unreasonable. If the neural network is a continuous function, its differential should be defined. Where we don't consider fractal functions. The explicit differential representation is published [3]. Then, function characters near discrete points of learning data are evaluated by the differential. The fact makes a discrete function into continuous; at least, an index for predictions is got.

2. Multi layer neural networks

We used a neural network without feedback loop, whose structure is three layers. In the layer structure, informed propagations are following.

$$\{x_1, x_2, \cdots x_n\} \equiv X,$$

$$y_j = \sum v_{ji} x_i, p_j = f(y_j),$$

where a vector {x1,x2,....xn} has an individual information, and its suffix corresponds to numbered neurons in the first layer. The neurons don't include bias ones. We substitute the bias neurons for threshold-values of actions of neurons. Suffices "i" and "j" are used for 1st and 2nd layer respectively. Vij is a matrix of connection weights between neurons in 1st and 2nd layer. The "yj" is a temporary variable. And f() is a function named neuron function to simulat a neuron, which must be differential function. If it is not differential, following learning equations are not defined. The "pj" is a vector for output of neurons in 2nd layer. Thus, we got following relations for informed propagations between 2nd and 3rd layers.

$$\{p_1, p_2, \cdots, p_m\} \equiv \boldsymbol{P}, z_k = \sum w_{jk} p_j$$

$$o_k = g(z_k), \{o_1, o_2, \cdots, o_k\} \equiv \boldsymbol{O},$$

Where Wjk is a matrix of connection weights, and g() is a neuron-function. Suffix "k" is used for 3rd layer, so a vector {o1,o2,...ok} is output for neurons in 3rd layer. In the neural network, an individual information {x1,x2,....xn} is transformed into a vector {o1,o2,...ok}. Where dimension of the vector is converted into different one, therefore, we must take care of the transformations in cases of "n < k" and large "m"-values. These relations are realized in one individual input/output datum, and the relations also stand up in case of plural data.

We arrived at following differential coefficients between output and input. For 3-layer-network:

$$\delta y_j = v_{ij} \delta x_i,$$

$$\delta p_j = f(\delta y_j) = f'(y_j) v_{ij} \delta x_i / 1!$$

$$+ f''(y_j)(v_{ij} \delta x_i)^2 / 2!$$

$$\delta z_k = w_{jk} \delta p_j,$$

$$\delta o_k = g(\delta z_k) = g'(z_k) w_{jk} \delta p_j / 1!$$

$$+ g''(z_k)(w_{jk} \delta p_j)^2 / 2!$$

$$\delta o_k = g'(z_k) \sum_j \left(w_{jk} \left(f'(y_j) v_{ij} \right.\right.$$

$$\left.\left. + f''(y_j) v_{ij}^2 \delta x_i / 2 \right)\right) \delta x_i$$

$$+ \frac{1}{2} g''(z_k) \sum_j \left(w_{jk}^2 \delta p_j \left(f'(y_j) v_{ij} \right.\right.$$

$$\left.\left. + f''(y_j) v_{ij}^2 \delta x_i / 2 \right)\right) \delta x_i$$

Consequently,

$$\frac{\delta o_k}{\delta x_i} = g'(z_k)\sum_j \left(w_{jk} f'(y_j) v_{ij}\right)$$

$$+\frac{1}{2} g''(z_k)\sum_j \left(w_{jk} f'(y_j) v_{ij}\right)^2 \delta x_i + \ldots\ldots$$

In ANN [4], absolutely $g''() = 0$, then:

$$\frac{\partial o_k}{\partial x_i} = \sum_j w_{jk} g'(q_j) v_{ij} f'(y_j),$$

For 3-layer multi infinitesimals:

$$\delta y_j = \sum_i v_{ij} \delta x_i$$

$$\delta p_j = f(\delta y_j) = f'(y_j)\sum_i v_{ij}\delta x_i$$

$$+\frac{1}{2} f''(y_j)(\sum_i v_{ij}\delta x_i)^2$$

$$\delta z_k = \sum_j w_{jk}\delta p_j$$

Therefore,

$$\delta o_k = g(\delta z_k) = g'(z_k)\sum_j w_{jk}\delta p_j$$

$$+\frac{1}{2} g''(z_k)(\sum_j w_{jk}\delta p_j)^2$$

In ANN or 1st. order approximation, absolutely $g''() = 0$, then:

$$\delta o_k = g'(z_k)\sum_j (w_{jk} f'(y_j)\sum_i v_{ij}\delta x_i)$$

Where f' and g' are differentials for f and g functions.

The neuron-functions f and g are sigmoid functions generally. But it is desirable that the functions are replaced by other ones. We often replace g-function with linear function. Such neural network is called ANN, which is excellent to extrapolate on various phenomena.

3. Recurrent representation of functions

We investigate a recurrent representation of functions in order to transform explicit periodic conditions into implicit. The representation is gotten from following facts,

(1) Sampling values of function are finite vector {x1,x2,....,xn}.

(2) A set of partial vector of the vector is written as {{x1,x2,...xk},{x2,x3,...xk+1},.....}, where an index k is less than n. "k-1" is called a window width. The index k has not meaning of neurons in third layer here. The width is determined by following index E,

$$D_{ij} = [\sum_l (x_{i+1} - x_{j+1})^2]^{0.5},$$

$$E = \min\{D_{ij}; i > j\}.$$

We used a condition that E is bigger than 0.02. The condition was got empirically. If the condition is negligible, the back propagation learning will be very difficult. If the learning may be completed, the generating potential plane is steep, and the steepness connects to the over learning for the neural networks.

(3) Elements of the set can be corresponded to a scalar series {xk+1,xk+2,...} as following: {{x1,x2,...xk}<-->xk+1,{x2,x3,...xk+1}<-->xk+2,...} They are set of partial vector and scalar series, and fragmentations of functions, which are learning data set for a neural network simultaneously. A property of neural networks combines a vector with another, which includes a scalar. That is {xi, xi+1,...xi+k-1}-->xi+k, where i=1,2,.... The relation is equivalent to a local prediction or fragmented forecasting. If the last partial vector is written as {xj, xj+1,...xj+k-1}, and the corresponding scalar is xj+k. In that case, what kind of character does the vector {xj+1, xj+2,...,xj+k} have? The {xj+1,xj+2,...xj+k} is a new undefined vector, but it is also an input datum for neural network. Therefore, a new scalar is calculated from the input datum. Is the scalar written as xj+k+1? In generally, it cannot be allowed. But if the sampled function is periodic and the new vector is included in learning data set, it will be allowed. In such a situation, the calculated scalar is xj+k+1, and a new vector {xj+2,xj+3,...xj+k+1} is also used for advanced prediction. These iterative series describe a periodic function implicitly. We call the iteration as a recurrent representation for the function.

4. Precision for recurrent representation

4.1 Euclid distance

When sigmoid functions are adopted in neural networks, output responses of the neural network near learning points are roughly equal to that of the points. And Euclid difference increases monotonously with distances from the learning points. The fact suggests implicitly that precision for the recurrent representation can be estimated by the distance. But the distance would not be quantitative index, because of the monotonous increasing only. We cannot discover quantitative variance around the learning points. We are sure that Euclid distance is 0th order approximation for the precision. The explicit formula "d" for the 0th approximation is,

$$d = \min\{D_1, D_2, \ldots\ldots D_N\}.$$

Where each "Di" is Di={ Σj[(xj-xi)^2+(xj+1-xi+1)^2 +....+(xj+k-1 - xi+k-1)^2]}$^{0.5}$,

and {xi,xi+1,...xi+k-1} (i=1,2,..,N) is the learning data, and {xj,xj+1,.....,xj+k-1} (j=1,2,...,N) is an input point for prediction.

4.2 Differential distance

Revising the defect for the variance around the learning points, we use differential coefficients for neural network. By using analogical derivations for Taylor's expansion, we get following a scheme, { δ j,... δ j+k-1}={(xj-xi),(xi+1-xj+1),..,(xj+k-1- xi+k-1)}, where index "i" means the nearest point from the

input point. Vector representation is,

$$\delta = x' - x,$$

$$\Delta = (\partial o / \partial x) \cdot \delta / 1! + (\partial^2 o / \partial x \partial x) \cdot [\delta \times \delta] / 2! + \ldots,$$

where symbol "×" means outer (tensor) products for the vector. The Δ has not a character for the distance. So we transform it into,

$$\Delta' = \{ \sum_j [((\partial o_i / \partial x_j)(x_j - x_i))^2 + ((\partial o_{i+1} / \partial x_{j+1})(x_{j+1} - x_{i+1}))^2 + \ldots\ldots + ((\partial o_{i+k-1} / \partial x_{j+k-1}) \times (x_{j+k-1} - x_{i+k-1}))^2]^{0.5}.$$

The scalar "Δ'" corresponds with Euclid distance. This scheme is valid on the learning points, but is invalid on far from the points. Such situations will be found during calculations to predict. So we are sure the index Δ' is 1st order approximation. Or, in a case of ANN, we use first order approximation,

$$\delta o_k = g'(z_k) \sum_j \left[W_{jk} f'(y_j) \sum_i [V_{ij} \delta x_i] \right],$$

$$\Delta'' = \left\{ \sum_k [\delta o_k^2] \right\}^{0.5}.$$

4.3 Multi dimensional prediction

The chaos is primary multi-dimensional phenomenon. We believe that it is multi-dimensional in deterministic chaos calculated numerically. Then, it should be expressed by multi descriptors. In the approximation of ANN, it is,

ok= Σj[Wjk*f(Σi[Vji*xi])]= Σj[Wjk*F(xi)].

The expansion is similar to the ortho-normal expansion. Since the function F is represented by sigmoid functions and their thresholds, the Fs are not ortho-normal set exactly. In order to close the Fs to the set, radial function's approach is often tried. It has been believed that methods similar to the ANN are useful. But they are not complete. If a phenomenon is described by 3 variables that is {x,y,z}, we will derivate following expressions,

ok= Σj[Wjk*F(xi)+W'jk*F(x'i)+ W"jk*F(x"zi)].

Where descriptors {xi,x'i,x"i} are correspond to variables {x,y,z}. In the expression, cross terms such as xi*x'i are not found. The expression is insufficient. We are reluctant to put the cross terms in first layer. That method adds new neurons that don't connect to outer region and don't reflect observations directly. But cross terms are necessary, we assume them to be {(x**a)*(y**b)*(z**c), a=0,1,2,..,b=0,1,2,..,c=0,1,2,}. The bias neuron exists in the assumption, that is a=b=c=0.

We must define a window width in order to part observation data into many fragments. The window is not for one dimension, but for the multi descriptors. But, we need to adopt a definition that includes one dimension domain. We resulted following correspondence, which are:

For x-coordinate, {{x1,x2,...xk}<-->xk+1, {x2,x3,...xk+1}<-->xk+2,....},

For y-coordinate, {{y1,y2,...yk}<-->yk+1, {y2,y3,...yk+1}<-->yk+2,....},

For z-coordinate, {{z1,z2,...zk}<-->zk+1, {z2,z3,...zk+1}<-->zk+2,....},

For any cross-term, {{r1,r2,...rk}<-->rk+1, {r2,r3,...rk+1}<-->rk+2,....}, where the letter "r" is a genetic symbol of (x**a)*(y**b)*(z**c)-term.

5. Forecasting of Ikeda's chaos

We have studied long-range forecasting for Resslor's and Lorenz' chaos. Because there is something like as "sine"-waves in the movements, which is similar to a sigmoid-function in the neuron, we think it is possible to simulated it by using neural-networks, and got reasonable simulated results [5]. On this work, we tried the Ikeda's chaos, which is well known as a laser-model [6], and the generate expression is simple. The definition of Ikeda's chaos is,

$$x_{n+1} = a + b(x_n \cos z_n - y_n \sin z_n)$$

$$y_{n+1} = b(x_n \sin z_n + y_n \cos z_n)$$

$$z_n = 0.4 - 6/(1 + x_n^2 + y_n^2)$$

where a and b are constant that we set 0.4 and 0.9. Initial values x0 and y0 were 0.0 and 0.0. We got figure 1 from the parameters.

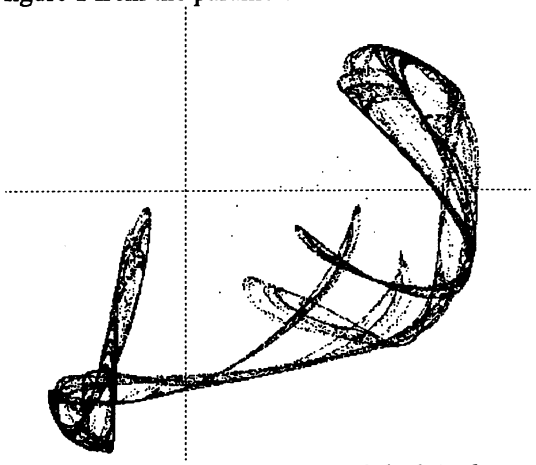


Figure 1. *Plotting of x, y-parts of Ikeda's chaos.*

Figure 1 shows a chaotic character. We got three kinds of time-series data sets {xi}, {yi}, and {zi}. The {zi} is not equivalent to others, which is seemed to be temporary-data in order to generate x- and y-parts. At first, we tried forecasting on using x-, y-, and z-parts as multi descriptors. But, we couldn't any acceptable results, even if many windows widths were used. We believed that the movements of Ikeda's chaos were too complex and had many bifurcations. So we used a method of the auxiliary function [7]. The method has several defects, and must be adopted under limited-conditions. But, if we wished to get reasonable forecasting, the method would be adopted. We tried forecasting of only x-part by using an auxiliary function that is y-part. Another auxiliary function was constant zero, which had no

function. Anyone will be sure that the forecasting is easy. The results are listed figure 2 and 3, which indicate possibility within very short-range forecasting, but as for long-range it is never acceptable.

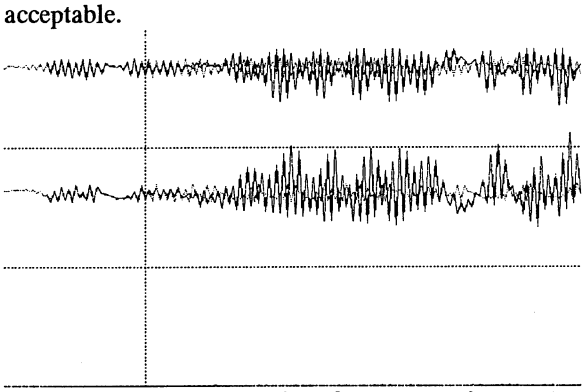


Figure 2. *The first graph is long-range forecasting for x-part of Ikeda's chaos. Where y-part in second graph is used as the auxiliary function. Third graph is constant zero, and it is used a kind of the auxiliary function.*

The left side of a dotted vertical line means a learning term (40 points). The right side is forecasting. Effective number of the learning data is 28, and forecasting points are 120. The solid curve is results from a neural network, and the plotted curve is true values of the chaos.

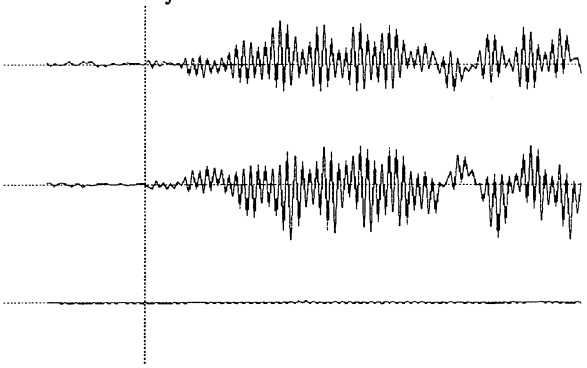


Figure 3. *Differences between output from neural network and the true value. From upper to bottom, they are x- , y-parts, and constant zero.*

Next, we used y- and z-parts as the auxiliary functions, and predicted x-part. The calculated results are listed in figure 4 and 5. These are not so satisfied for us, but as concerning complex movements of Ikeda's chaos, they would be one upper limit for forecasting method by using neural networks.

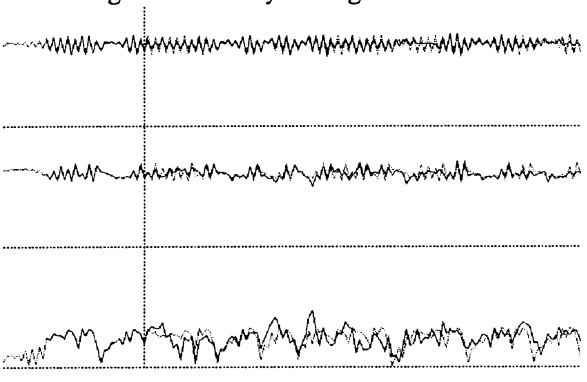


Figure 4. *First graph is long-range forecasting for x-part of Ikeda's chaos. Second and third graphs are y- and z-parts of the chaos, and they are used as the auxiliary functions. Where learning points are 40, and forecasting ones are 120.*

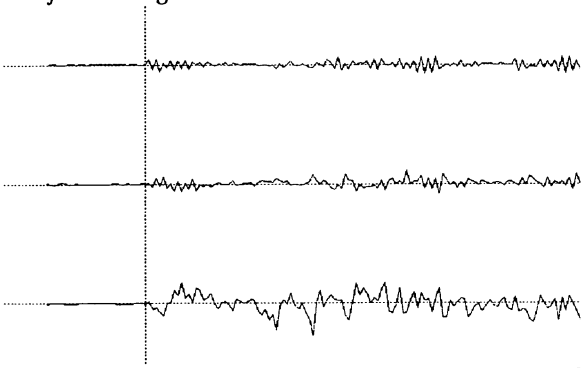


Figure 5. *Differences between output from neural network and the true value. From upper to bottom, they are x- , y-, and z-parts of Ikeda's chaos.*

6. Conclusion

We investigated differential coefficients for multi neural networks, and derived an index to relate to a precision of forecast. We discussed recurrent representations of functions, and which were used in the processing of neural networks. We examined these processing techniques on the forecasting of Ikeda's chaos. The method used by us has high accuracy and long-range forecasting facility. It is a practical method. However, in the examinations for Ikeda's chaos, a limitation is pointed out. It would be an original lack arisen from the pattern recognition.

References

[1] I.Matsuba,"Chaos and Forecasting",Mathematical Science (in Japanese), pp.64 (Science Co. Ltd., Tokyo, 1992.6).

[2] M.Onodera, U.Nagashima, H.Yoshida, T.Aoyama, and H.Hosoya, "Neural Network Reproduction of Time Series Data with Varying Amplitudes and Frequencies (in Japanese)", J. Chemical Software, Vol.4, pp.57(1998).

[3] T.Aoyama, Y.Isu, and U.Nagashima,"Extra-polations by using neural-networks and a recurrent representations of functions (in Japanese)", IPSJ SIG Notes, 95-HPC-59(1995.12.11).

[4] T.Aoyama, Y.Suzuki, and H.Ichikawa, "Neural Networks to quantitative structure-activity-relation-ships", J.Med.Chem, 33, 2583(1990).

[5] H.Zhu, T.Aoyama, and I.Yoshihara,"Forecasting and precision on using multi-layer neural network", Proceedings of the 14th KACC, E-218 (1999.10.14).

[6] K.Ikeda,"Multiple-valued stationary state and its instability of the transmitted light by a ring cavity system", Opt. Commun., Vol.30, pp.257-261, 1979.

[7] T.Teshima, H.Zhu, T.Aoyama, and I.Yoshihara, "Forecasting and precision on using multi-layer neural networks: Introducing of auxiliary functions". AROB 5th.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Characteristics of an adaptive associative memory system based on autonomous image reaction

Yasuo Kinouchi*, Naoki Matsumoto*, Masahiro Mizutani*, and Shoji Inabayashi**
* Tokyo University of Information Sciences, 1200-2 Yatoh-cho Wakaba-ku Chiba-shi 265-8501 Japan
** Pacific Systems Corp., 8-4-19 Tajima Urawa-shi 336-0037 Japan

Abstract

Characteristics of an adaptive associative memory system based on autonomous image reaction are investigated by computer simulation with a virtual robot. This memory system constitutes a core part of a "thinking like" processing system, which we call "image reactor." The virtual robot in the computer has a brain composed of this adaptive associative memory system and adapts to a virtual environment (miniature garden) as it searches for food in a computer simulation. While the robot walks around the miniature garden, the network structure of images in the memory system is modified according to selections and generations based on autonomous reactions between images.

1. Introduction

To construct a new computer which works like "thinking," we are developing an adaptive associative memory system. This system treats "images" as basic units of information and adapts to an environment of the external world by the autonomous reactions between the images and generate prototype images of the external environment[1][2]. In this paper, the characteristics of the memory system are investigated by computer simulation with a virtual robot[3]. The virtual robot has a brain incorporating this memory system and adapts to a virtual environment (a miniature garden) as it searches for food.

Scenes of the miniature garden from the robot's perspective are input in the associative memory system as images one by one at each step as the robot "walks" through the garden. First, Based on scenes that the robot is seeing at the time, the memory system calls up or "remembers" an image which has already been memorized or has been generated from past scenes. Next, the memory system recalls the direction from the remembered scenes. The robot then walks in the direction that its memory system recalls. While robot walks around the miniature garden, network structure of images in the memory system are modified according to selections based on Hebbian rule and reinforcement learning method. The memory system also generates prototype images through autonomous reactions between images and natural selection of images. Images having high action potential autonomously and randomly react to each other to generate new images and links. These reactions are, in a sense, a primitive form of "concept generation."

This paper describes the configuration and behavior of the robot and memory system and shows the results of the computer simulation. The total number of images which were input or generated in the memory system was over 300,000. Through the process of image selection, about 400 to700 images on average exist in the memory system.

2. Basic concepts

2.1 Basic concepts of "thinking like" processing system

This research is carried out under following concepts, for constructing a "thinking like" processing system.
(1) A "thinking like" processing system itself autonomously adapts to the environment. Such a processing system can be composed by integrating memory and information processing functions. As an integrated processing system, the memory system makes an image, using local representation, as a memory network component.
(2) The memory system adapts to the environment by generating and selecting images. Images are randomly generated based on autonomous reactions between images. However, these reactions are not generated completely randomly. The links from one image to another that control the association works as conditions of constraint.
(3) The selection mechanism of images is based on Hebbian rule, and selection of images is controlled by the level of action potential that the image has.
(4) The memory system generates prototype images through autonomous reactions between images and selection of images. These reactions are, in a sense, a primitive form of "concept generation."
(5) The generated prototype images are strongly connected in the memory system. This strongly connected network, which is generated as a result of adaptations of the processing system, represents processing rules that are effective for the specific environment.

2.2 Image as information-representation units

In many cases, images or patterns have the following properties.
(1) Separability: Images are "separable" into smaller images that show a part or a feature of the larger image, but that can still be identified as a part or a feature of the larger image.
(2) Connectivity: Connectivity is the inverse property of separability. New images can be created by spatially combining several images into a unit.
(3) Stability (Redundancy): The meaning of what is being represented does not change much if its form changes within a certain range. That is, a minor change in an image does not change its meaning very much.

In this paper, information that satisfies the above properties is treated as an image in a broad sense, although at first glance it may not seem like an image. Moreover, in the following descriptions, the image is represented as a set of features or pseudo-features from next reasons.
(1) Set of features or pseudo-features of images obviously satisfies the three properties described above.
(2) By expressing the image as a set of features or pseudo-features, the reaction between images can be simply defined as an operation on the existence of features of each image.

This system uses local representation, which means that an image is stored in a node. This method of representation allows the content and association (access) method of the memory to easily changed independently. The advantage of enabling the memory to be changed independently is that large-capacity associative memory systems can be easily set up.

3. Configuration and function

3.1 Basic configuration of image network

The basic elements of the image memory network are nodes and links. Each node contains one image and is connected to other nodes by links which are weighted. The image data is expressed as an array of digital bits ones and zeroes. A "one" shows that a pseudo-feature is included in the image. A "zero" shows that a pseudo-feature is not included in the image.

The condition of a node is defined by the level of its action potential and whether the node is excited. The action potential shows the degree of activity of the node, and it changes between "zero" and "one". An excited node transmits stimulation signals in one direction to other nodes. The product of the individual excitement signal and the links weight becomes the input of other node. Each individual node has a threshold which determines whether it is excited.

The associative memory's adaptability is realized by the autonomous reactions of the node. The autonomous reaction of an image generates new images from old images. At the same time old images with low action potentials disappear, and images that resemble other images merge into one image. By these process of image generation, disappearance, and combination make up the natural selection of images for the whole network. Through this natural selection, the memory system itself adapts to the external environment.

3.2 Basic configuration and processing functions

Basic configuration of the image reactor are shown in Fig.1. The image reactor is composed of a preprocessing module, a feature-extraction module, a reaction module, a motor module, and a control module.

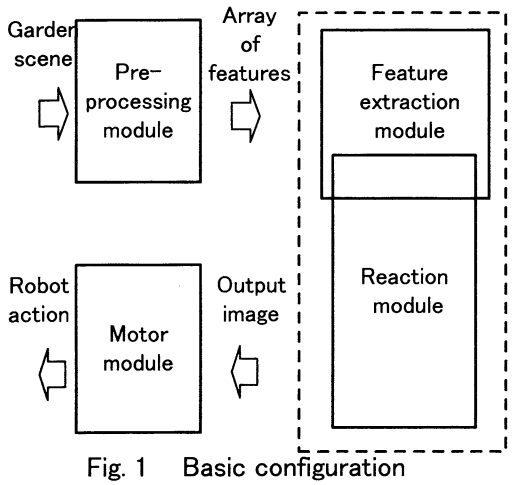


Fig. 1 Basic configuration

A flow of processing functions is shown below.

(1) Pre-processing stage: In the first stage, the pre-processing module takes image information from external environment. In this pre-processing module, image of scenes of external environment are converted into the array of pseudo-features.

(2) Feature-extraction stage: In the next stage, the feature-extraction module detect groups of pseudo-features from the input image of the pseudo-feature array. The feature-extraction module has a large number of nodes for detecting features which is done by evaluating the resemblance between the input image data array and its own image data array. If the resemblance is high, these feature extraction nodes become excited, and send a stimulation signal through the link.

There are two types of images in the feature-extraction module. One type is included only in the feature-extraction module and the other type is included in both the feature-extraction module and the reaction 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 for both previously prepared images and images for which the action potential is high.

(3) Association stage: Images in the reaction module receive signals from multiple feature extraction nodes and calculate the total value of the stimulation signals, incorporating the weight of the link in the calculation. Depending on the result, the node may or may not become excited. Nodes having image data that are similar to the input image data are more easily excited. Excited images transmits stimulates signals to other nodes in the reactor module. In the association stage, the points of each node is calculated as

Node point = Total weighted value of the stimulation signals $\times$ action potential

The image with the most node points is assumed to be a recalled image according to the associative memory. The output of the associative memory is the image with the most node points.

(4) Decision stage: The robot can move in four directions only: north, east, south, and west. The direction is determined stochastically according to the weight set of motion which are connected to recalled image. Each node in the reactor module is connected to each weight set of motion which determine probability of moving in a certain direction. The robot continue walking around until it finds food or poison. The weight set changes according to the rules of reinforcement learning, as will be shown next.

(5) Learning stage: The robot stops walking around when it finds food or poison. At this point the robot corrects the weight set of motion and action potential based on the reinforcement learning. The weights are increased for the direction in which the robot moved if the robot found the food. Similarly, the action potential of images recalled during the robot's movement is increased, when the robot finds the food. In other words, the images which are connected with good results are memorized.

(6) Reaction stage: The images in the reaction module generate new nodes and links based on the autonomous reaction between nodes. Images with high action potential

mutually and randomly react to each other as shown in Fig.2. The input to a reaction is a set of two images and their links. The output is a new image and new links to/from the new image from/to 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 simultaneously reacting images is large.

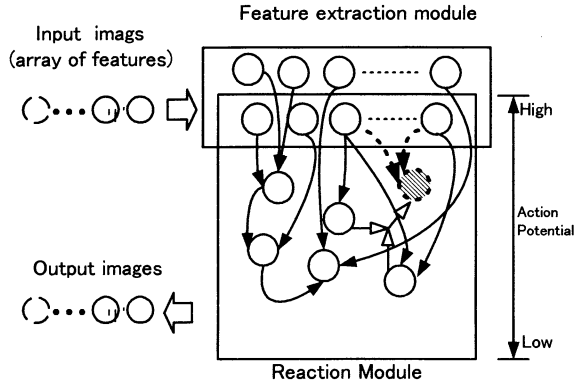


Fig.2 Image reaction

3.3 Image reaction

Array of pseudo-features are assumed here to be subject to three basic operations called Addition, Multiplication, and Mergence. Operations between the input image data Ia and Ib of nodes A and B generate output node C with image data Ic where image data Ia, Ib, and Ic are array of features. These operations are defined as follows.
Addition: Ic = Ia + Ib.
Ic has the features of both Ia and Ib.
Multiplication: Ic = Ia * Ib.
Ic has features common to both Ia and Ib.
Mergence: If Ia $\fallingdotseq$ Ib, then Ic = Ia + Ib.
Ic has features of both Ia and Ib, and Ia and Ib disappear.
The links to/from the node C from/to others is created as follows.
Addition: Only links connected to A or B are linked to C.
Multiplication: Links connected to only both A and B are linked to C.

Addition and Multiplication are executed only when the direction of the transfer vectors of the two arrays are nearly the same. The transfer vector shows the average change of the robot position per a step which is stochastically calculated from the action weight of image. This restriction ensures that Addition and Multiplication are performed only on those images whose action will result in almost the same motion. By imposing this restriction, the group of pseudo-features considered to have a similar meaning in a series of experiences is extracted.

Addition generates composite and intermediate images, and Multiplication is a kind of abstraction from concrete images. Mergence eliminates similar images. These three operations cooperate in selecting images, so the image network constitutes a kind of concept-generation model.

3.4. Selection and reinforcement learning

The image reactor changes the action potential and weights of links based on Hebbian rule, and learns based on reinforcement manner. The action potential of a node increases when the node is excited and decreases when it is not. A node (image) disappears, when its action potential drops below a threshold. The weight of a link approaches zero when stimulation signals do not propagate through it. Nodes disappear at a rate almost equal to the rate at which new images are generated.

When the robot eats food the action potential of a node increases, and when robot takes poison, the action potential of a node decreases. These increase and decrease are based on the profit-sharing method.

4. Simulation system

4.1 Configuration of the simulation system

The robot observes the scene in 5 directions simultaneously, and at each step walks in any of 4 directions. The miniature garden is a 10×10 lattice of 8×8 bit patterns representing scenes and figures. For the robot, patterns are also set of pseudo-features. Since the input is already a set of pseudo-features, the pre-processing module does nothing. The robot starts from the designated start point, and scenes observed in each direction are input, image associations are carried out, the robot takes one step, and the operation from the input of the scene to the action decision the ("step operation") is repeated. From the robot starting to the robot finding food or poison, learning, and image reaction is what we call the "cycle operation." When one cycle operation ends, the robot is returned to the start point, and a new cycle operation begins. The condition of the images in the reactor module is maintained when the robot begins the new cycle operation. That is, memory persists from one cycle operation to the next.

4.2 Configuration of the simulation program.

The simulation program was written in C++. The number of images fluctuated drastically while the simulation was running, but the number of images and links could be changed dynamically without limit by using the pointing operation in the program. The spreadsheet software was used to analyze the log-out data off line, and log-out of the conditions of images and links in the image reactor were monitored at every step for the following reasons.
(1) The particular parameters needing to be monitored could not be identified beforehand. The use of spreadsheet software enabled various parameters to be evaluated off line.
(2) Step-by-step monitoring made this very complicated simulation program easy to debug.

4.3 Simulation result

An example of the number of the active images during 1000 cycles is shown in Fig.3. This figure shows that the average number of active images is about 400 to700. Fig.4 shows the total number of the image which are input and generated by the reaction, the number of merged images, and the number of disappeared images. About 350 images were generated averagely at each cycle, and many

images merged. In this simulation, rate at which the similar images were generated was usually high, and the number of the images which fading away by the lowering of their action potential was small: about 10 per cycle.

The relation between action potential and the number of reactions of each image after 1000 cycles is shown in the Fig.5, we can see that few images had high action potential and frequent reaction. We think that these images function as a primitive concept.

But the frequency with which the robot found food did not increase with time. In this program, the image selection was based on the difference between poison and food: the image seen when eating poison was forgotten, and the image seen when eating food was memorized resistant. The robot thus "forgot" the scenes associated with the poison and consequently tended to eat the poison frequently. We think that the images associate with dangers should be left. Image selection methods that enables the action which positively avoids the danger are necessary.

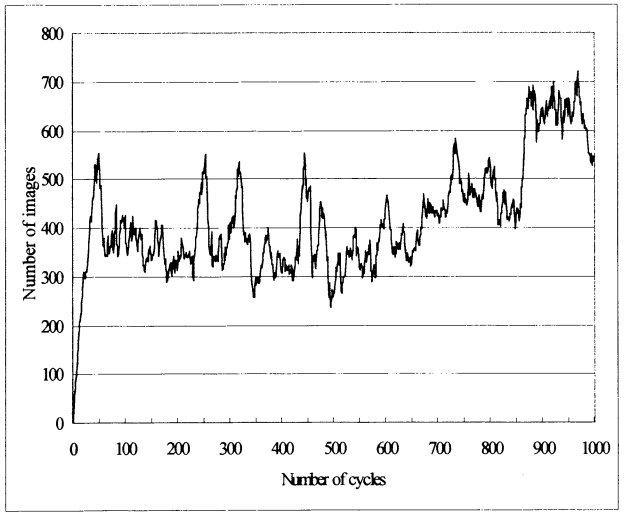


Fig.3 Number of active images

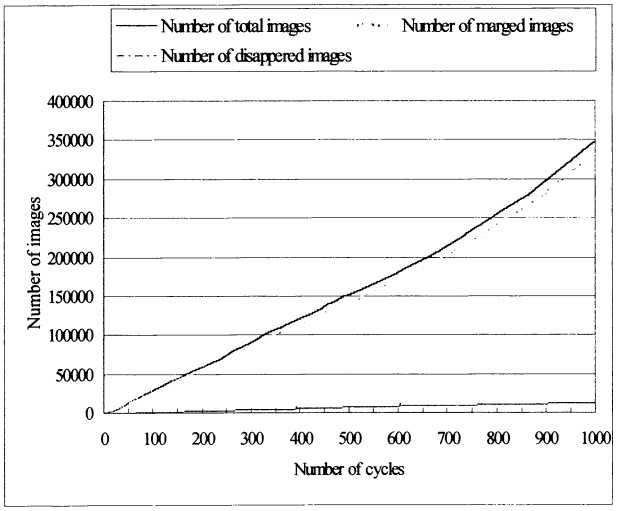


Fig.4 Number of total, marged, and disappeared images

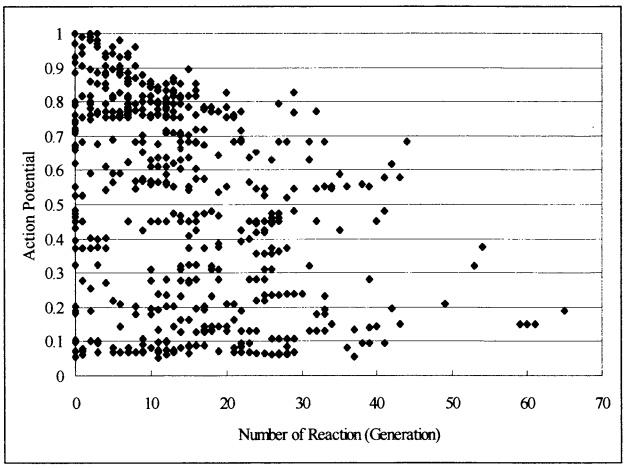


Fig.5 Relation between action potential and number of reaction

5. Conclusion

We have described the characteristics of an associative memory system with an adaptation function that can be used to construct a "thinking-like" processing system and have described the simulation system used to evaluate this memory system. Using the simulator, we will further examine the characteristics of the image reactor, and will focus on how the image network "learns" and how the image "interact."

Acknowledgments

This work was supported by the High-Tech Research Center Project of the Ministry of Education, Science and Culture. We thank Prof. Shuji Hashimoto and Mr. Pitoyo Hartono of Waseda University for useful suggestions and we also thank Prof. Tadashi Watanabe, Dr. Kazuhisa Kitakaze, Dr. Kazuko Yamasaki, and Dr. Fumikazu Iseki of Tokyo University of Information Sciences for helpful discussions.

References

[1]Kinouchi Y, Mizutani M, and Inabayashi S,
An Architecture for an Adaptive Associative Memory System Based on Autonomous Reactions between Images. Proceedings of the International Joint Conference on Neural Networks '99 (No.774)

[2] Kinouchi Y, Mizutani M, and Inabayashi et. al. ,
An Adaptive Associative Memory System based on Autonomous Reaction between Image Memories.
Proceedings of the Fourth Int. Symp. On Artificial Life and Robotics (AROB 4th '99) (1999) pp.707-710

[3] Meyer A, and Guillot A, Simulation of adaptive behavior in animats: Review and Prospect From Animal to Animats. Proceedings of the First International Conference on Simulation of Adaptive Behavior(1991).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Optimization of Wavelet Filters for Image Compression by Using Genetic Algorithm

Jinshan Tang*
Beijing University of Posts and Telecommunications, Beijing,China

Jizhou Sun
Computer Department, Beijing Branch, Shanghai Pudong Development Bank,Beijing,China

Y. G. Zhang
Institute of Systems Science, Academia Sinica ,Beijing,China

Abstract: The problem discussed in this paper is devoted to the optimization of wavelet filters for image compression by genetic algorithm.

Key words image compression, optimal wavelet, genetic algorithm, gradient algorithm

1. Introduction

In wavelet based image compression techniques, the choice of wavelet filters is a key problem that affects image quality as well as system design. Although regularity was used in wavelet filters choice, its success at predicting compression performance is only partial[1]. In order to select good wavelets so that the compression performance is improved, optimal wavelet problem has drawn a lot of attention in the past years[2][3][4][5]. The basic idea of optimal wavelet problem is to look for prototype filters that optimize criteria such as energy compaction, aliasing energy and subband cross-correlation. So, in optimal wavelet problem, two problems are important. One is the choice of criteria, the other is the optimization algorithm. Especially after the criteria is fixed, the optimization algorithm is more important.

In the past years, gradient algorithms are used in the optimization of wavelet filters. However, because of their local convergence, the effectiveness is not good. Especially when the cost function is complicate, the algorithms will become worse. In order to improve the performance, this paper will study genetic algorithms[7] to deal with the optimization problem. Our experimental results prove that genetic algorithms are better than gradient algorithms. Genetic algorithms are not suitable for the comparatively simple cost function, but also suitable for complicate cost function.

The paper is organized as follows: Section 2 discusses image compression using wavelet filters and the optimal wavelet problem. Section 3 introduces genetic algorithms simply . Section 4 gives some experimental results and section 5 concludes the paper.

2. Wavelet transform and the optimal wavelet base

The wavelet transform of an image can be described in figure 1. Here

$$H_1(w)=\frac{1}{\sqrt{2}}\sum_{n\in Z}h_n e^{-inw} \text{ and } H_2(w)=\frac{1}{\sqrt{2}}\sum_{n\in Z}h'_n e^{-inw}$$

are low-pass filters.

$$G_1(w)=\frac{1}{\sqrt{2}}\sum_{n\in Z}g_n e^{-inw} \text{ and } G_2(w)=\frac{1}{\sqrt{2}}\sum_{n\in Z}g'_n e^{-inw}$$

are high-pass filters such that

$$G_i(w)=e^{-jw}H_i^*(w+\pi)(i=1,2)$$

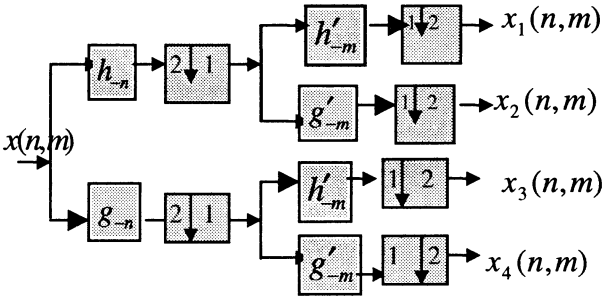


Figure1. wavelet transform

In wavelet transform, the choice of $H_1(w)$ and $H_2(w)$ is very important. Suitable $H_1(w)$ and $H_2(w)$ can bring a lot of advantages to image compression. There are many papers on the choice of $H_1(w)$ and $H_2(w)$. In this paper, we'll give a new method suitable for image compression. The objective measure of the efficiency of this method is given by:

$$L=E(x_1(n,m)-E(x_1(n,m)))^2 \quad (1)$$

where $E(.)$ is the expectation.

With a stationary random field, we can prove the following result:

$$L=G^*\int_{-\pi}^{\pi}\int|H_1(w_1)H_2(w_2)|^2 S_x(w_1,w_2)dw_1dw_2 \quad (2)$$

where $S_x(w_1,w_2)$ is the spectral density function of the field and G is a constant. Now, we'll consider the design of the optimal wavelet base.

* Now the author is working in ATR Media Integration & Communications Research Laboratories, Seika-cho, Soraku-gun, Kyoto, 619-02 Japan Phone: +81-774-95-1429 FAX: +81-774-95-1408 E-mail: stang@mic.atr.co.jp

Note that $H_1(w)$ and $H_2(w)$ are completely determined by the sequence $\{h_k\}$ and $\{h'_k\}$. Thus, finding the optimal $H_1(w)$ and $H_2(w)$ is really equivalent to finding the optimal $\{h_k\}$ and $\{h'_k\}$. But the minimization of (2) is very difficult, because it is a constrained optimization problem with constraints over sequence $\{h_k\}$ and $\{h'_k\}$. In order to overcome the difficulty, we have to use parametrization technique. After parametrization, sequence $\{h_k\}$ and $\{h'_k\}$ can be written as the function of $\{\theta_k\}$ and $\{\theta'_k\}$, i.e $h_k = h_k(\theta_1, \theta_2, ..., \theta_m)$ and $h'_k = h'_k(\theta'_1, \theta'_2, ..., \theta'_{m'})$. Thus the optimization problem over the constrained sequence $\{h_k\}$ and $\{h'_k\}$ can be reformulated as an unconstrained problem over the constrained sequence $\{\theta_k\}$ (k=1,2,...,m) and $\{\theta'_k\}$ (k=1,2,...,m') where θ_k and θ'_k belong to $[0, 2\pi]$. With more details, the readers can refer to [6].

Let

$$\vec{h}(\vec{\theta}) = \begin{pmatrix} h_0(\vec{\theta}) \\ h_1(\vec{\theta}) \\ \vdots \\ h_{2N_1-1}(\vec{\theta}) \end{pmatrix}, \quad \vec{h'}(\vec{\theta'}) = \begin{pmatrix} h'_0(\vec{\theta'}) \\ h'_1(\vec{\theta'}) \\ \vdots \\ h'_{2N_2-1}(\vec{\theta'}) \end{pmatrix}$$

where $\vec{\theta} = \begin{pmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_m \end{pmatrix}$ and $\vec{\theta}' = \begin{pmatrix} \theta'_1 \\ \theta'_2 \\ \vdots \\ \theta'_{m'} \end{pmatrix}$.

Then

$$|H_1(w_1)|^2 = H_1(w_1)\overline{H_1(w_1)} = h^T(\vec{\theta})A_1(w_1)h(\vec{\theta}) \quad (3)$$

$$|H_2(w_2)|^2 = H_2(w_2)\overline{H_2(w_2)} = h'^T(\vec{\theta'})A_2(w_2)h'(\vec{\theta'}) \quad (4)$$

Where

$$A_1(w_1) = \begin{pmatrix} 1 \\ e^{-jw_1} \\ \vdots \\ e^{-jw_1(2N_1-1)} \end{pmatrix} \begin{pmatrix} 1 & e^{jw_1} & \cdots & e^{jw_1(2N_1-1)} \end{pmatrix}$$

$$A_2(w_2) = \begin{pmatrix} 1 \\ e^{-jw_2} \\ \vdots \\ e^{-jw_2(2N_2-1)} \end{pmatrix} \begin{pmatrix} 1 & e^{jw_2} & \cdots & e^{jw_2(2N_2-1)} \end{pmatrix}$$

Suppose that $S_x(w_1, w_2)$ is separable, i.e.,

$$S_x(w_1, w_2) = S_1(w_1)S_2(w_2) \quad (5)$$

Substituting it into (2),we obtain,

$$L = 4\pi^2 G * h^T(\vec{\theta})E_1 h(\vec{\theta})h'^T(\vec{\theta'})E_2 h'(\vec{\theta'}) \quad (6)$$

where

$$E_1 = \frac{1}{2\pi}\int_{-\pi}^{\pi} A_1(w_1)S_1(w_1)dw_1$$

$$E_2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} A_2(w_2)S_2(w_2)dw_2$$

Thus, the constrainted optimization problem is changed into an unconstrained optimization problem. And from(6), we can find that the two-dimensional optimzation problem can be changed into two one-dimensional optimization problems, that is, find $\{\theta_k\}_{1\le k\le m}$ such that

$$L_1(\vec{\theta}) = h^T(\vec{\theta})E_1 h(\vec{\theta}) \quad (7)$$

is maximized and find $\{\theta_k\}_{1\le k\le m'}$ such that

$$L_2(\vec{\theta}) = h'^T(\vec{\theta'})E_2 h'(\vec{\theta'}) \quad (8)$$

is maximized.

Before we begin to solve the two optimization problems, we must know the matrices E_1 and E_2. In order to obtain the two matrices, we must know the statistical models of the image sources, from which we can obtain the spectral density function of the field $S_x(w_1, w_2)$,which is needed in the optimization problems.

In image compression, the images are generally modeled as a 2D random stationary field, denote by $x(n_1, n_2)$ .The autocorrelation function of the random field $x(n_1, n_2)$, $R(n_1, n_2; m_1, m_2)$,is defined by

$$R(n_1, n_2; m_1, m_2) = E[x(n_1, n_2; m_1, m_2)] \quad (9)$$

The autocorrelation is the expectation of the product of the two random variables $x(n_1, n_2)$ and $x(m_1, m_2)$ .A random field $x(n_1, n_2)$ with nonzero mean can always be transformed to a zero mean random field by subtracting $E[x(n_1, n_2)]$ from $x(n_1, n_2)$. So, in the paper, unless specified otherwise, we will assume that $x(n_1, n_2)$ is a zero-mean field. We know that the autocorrelation of the stationary field $x(n_1, n_2)$

depends only on the lag $m_1 - n_1, m_2 - n_2$. So, we can rewrite(9) as follow:

$$R(n_1,n_2) = E[x(m_1,m_2;m_1-n_1,m_2-n_2)] \quad (10)$$

for all(m_1, m_2).

The spectral density function of a stationary random fields $x(n_1,n_2)$, $S_x(w_1,w_2)$, is defined by:

$$S_x(w_1,w_2) = \sum_{-\infty}^{\infty}\sum_{-\infty}^{\infty} R_x(n_1,n_2)\, e^{-j(w_1 n_1 + w_2 n_2)} \quad (11)$$

In image processing, generally we adopt the separable model, the autocorrelation of which is separable, i.e.

$$R(n_1,n_2) = R(n_1)\ R(n_2) \quad (12)$$

This type often used in image processing is

$$R(n_1,n_2) = \sigma^2 \rho_1^{|n_1|} \rho_2^{|n_2|} \qquad |\rho_1| < 1, \quad |\rho_2| < 1 \quad (13)$$

Here σ^2 represents the variance of the random field and $\rho_1 = R(1,0)/\sigma^2, \rho_2 = R(0,1)/\sigma^2$ are the one-step correlations in the x and y directions, respectively.

Substitute (13) into (11), we obtain the spectral function as follows:

$$s_x(w_1,w_2) = \frac{\delta^2(1-\rho_1^2)(1-\rho_2^2)}{(1+\rho_1^2-2\rho_1\cos w_1)(1+\rho_2^2-2\cos w_2)} \quad (14)$$

Let

$$S_1(w_1) = \delta_1^2 \frac{1-\rho_1^2}{1+\rho_1^2-2\rho_1\cos\ w_1} \quad (15)$$

$$S_2(w_2) = \delta_2^2 \frac{1-\rho_2^2}{1+\rho_2^2-2\rho_2\cos\ w_2} \quad (16)$$

where $\delta^2 = \delta_1^2 * \delta_2^2$.

By using (15) and (16), it is easy to obtain E_1, E_2.

3. Genetic algorithm

Genetic algorithms are powerful stochastic and optimization techniques based on principles from evolution theory. Compared with many other optimization methods, genetic algorithms are proven more effective in multi-peak optimization problems[7]. They can overcome some disadvantages in some optimization methods such as gradient algorithms. We know that, in the optimization methods such as gradient algorithms, we need to move from a single point in the decision space to the next using some transition rules to determine the next point. So these point-to-point methods are not reliable in multi-peak optimization problems, which often converge to local maximum points. In contrast, Genetic algorithms can overcome these disadvantages by climbing many peaks in parallel.

The basic steps of genetic algorithm can be described as follows[7][8]:

Step1. Initialize a population of chromosomes
Stpe2. Evaluate each chromosome in the population
Stpe3. Create new chromosomes by current chromosomes with reproduction, crossover and mutation.
Step4. Delete members of the population to make room for the new chromosomes
Step5. Evaluate the new chromosomes and insert them into the population.
Step6. If time is up, stop and return the best chromosome; if not, go to step3.

4. Experimental results and discussion

In our experiments, we only consider 4-tap orthogonal wavelet filters , other cases can be studied in the same way.

4.1. Parametrization of 4-tap orthogonal wavelet filters

By using parametrization technique, the 4-tap wavelet filters can be described as follows[9]:

$$h_0 = \tfrac{1}{\sqrt{2}}(\sin^2\theta_1 + \sin\theta_1\cos\theta_1)\ h_1 = \tfrac{1}{\sqrt{2}}(\sin^2\theta_1 - \sin\theta_1\cos\theta_1)$$

$$h_2 = \tfrac{1}{\sqrt{2}}(\cos^2\theta_1 - \sin\theta_1\cos\theta_1)\ h_3 = \tfrac{1}{\sqrt{2}}(\cos^2\theta_1 + \sin\theta_1\cos\theta_1)$$

After parametrization, h_k is a function only including one variable.

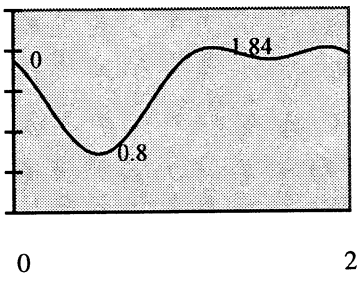


Figure2. The shape of function $L_1(\theta_1)$

4.2. Shape of the function $L_1(\vec{\theta}) = h^T(\vec{\theta})E_1 h(\vec{\theta})$

In order to simplify the problems, the following processing is done:

(1) $\delta_1^2 = 1$; (2) replace h_i by $\frac{1}{\sqrt{2}}h_i$; (3) $\rho_1 = 0.95$

From [9], we know that the above assumption doesn't affect the solutions of (7).

Figure2 describes the shape of the function $L_1(\theta_1)$. From figure 2,we know that it is a multi-peak function, so general gradient techniques can not obtain good results.

4.3. Comparison of optimization results by using gradient algorithm and by using genetic algorithm

In order to compare the performance, in our experiments, we use gradient algorithm and genetic algorithm to solve the optimization problem respectively.

The experimental results of gradient algorithm are listed in table 1. When we use gradient algorithm to do the experiments, 1.0, and 3.0 are chosen as the original values. From table 1, we can understand that different choice of original values results in different solutions to the optimization problem. The reason is that gradient algorithm is locally convergent. If the choice of original value is not good, it is impossible to find the optimal solutions.

Table 1. optimization results by using gradient algorithm

Original values θ_1	1.0	3.0
$L(\theta_1)$	0.934797942638	0.979662418
Local optimal value θ_1	0.785398	2.879793
$L(\theta_1)$	0.928687393665	0.9807890653

Table 2. optimization results by using genetic algorithm

No. of Experiments	NP	LC	PC	PM	NG	θ_1	fitness
1	6	10	0.95	0.07	2000	1.867144	0.98070
2	6	10	0.95	0.07	2000	1.8333636	0.9807890
3	10	15	0.95	0.07	2000	1.833450	0.9807891

The experimental results by using genetic algorithm are listed in table2. From table2, we can find that, although the solutions in different experiments are different, depending on the number of the population(**NP**), the number of the generation(**NG**), the probability of crossover(**PC**), the probability of mutation (**PM**), the length of chromosome(**LC**), every solution is close to the optimal solution θ_1 =1.832596 [9].

4.4. Experimental results of image compression

In this section, experimental results of image compression by using the optimal wavelet obtained by genetic algorithm and the wavelet chosen randomly are given. For the tests, the standard 512× 512 test image Lena was used. The wavelet decomposition consists of four levels. The image quality is measured by the peak signal-to-noise (PSNR) and compression ratio(Cr), computed from actually decoded images. The compression algorithm developed by us[9] is used in our experiments.

The optimal wavelet filter obtained by genetic algorithm is

θ_1 = 1.833450;

h_0=0.3409223$\sqrt{2}$, h_1=0.5916623$\sqrt{2}$,

h_2=0.1590777$\sqrt{2}$, h_3 =-0.0916623$\sqrt{2}$;

and the wavelet chosen randomly is

θ_1=0.8;

h_0=0.5071993$\sqrt{2}$, h_1=0.007406486$\sqrt{2}$,

h_2=-0.007193287$\sqrt{2}$, h_3 =0.4925935$\sqrt{2}$;

Figure 3 shows the performance of the two wavelet filters. From Figure3, we know the performance of the optimal wavelet filter is much better than the wavelet chosen randomly.



(a) θ_1 =1.8334506750115
PSNR=33.013794
Cr=14.5280425



(b) θ_1 =0.8
PSNR=24.962715
Cr=13.563615667

Figure 3. Decoded images

5. Conclusion

In this paper, genetic algorithm has been proposed for the design of the optimal wavelet base in image compression. Compared with gradient algorithms which sometimes fail to find the optimal solutions, genetic algorithm can always find the near optimal solutions.

References

[1] John D.Villasenor, Benjamin Belzer, and Judy Liao,"Wavelet filter evaluation for image compression",IEEE Tr. on Image Processing,Vol.4, No.8, August,1995.

[2] A.H.Tewfik, D,sinha, and P.Jorgensen."On the optimal chioce of a wavelet for signal representation",IEEE Tr. on Information Theory,Vol,38,No.2,pp 747-765,Mar,1992.

[3] Yan Zhang, John S. Baras,"Constructing optimal wavelet basis for image compression",ICASSP-96,2351-2354.

[4] Francois Moreau de Saint-Martin, Emmanuel Gautier, Jean-Pierre d'Ales de Corbet,"Signal-based design of wavelet filters", ISCASSP-96,1495-1498.

[5] Sze Fong Yau, Durgesh Srivastava,Shiu Wing Tso,"Image coding using generalized optimal subband decomposition and vector quantization",Signal Processing: Image Communication 8(1996)161-168.

[6] Hehong Zou and Ahmed H.Tewfik,"Parametrization of compactly supported orthonormal wavelet ,"IEEE Trans.signal processing,pp1428-1431,March,1993.

[7] Goldberg D. E. Genetic algorithm in search,optimization, and machine learning. Addison-wesley, 1989.

[8]Lawrence Davis(1991), "Handbook of genetic algorithms",Van Nostrand Reinhold,New York.

[9] Jinshan Tang,(1998) " Research of Several Important Problems in Wavelet-Transform-Based Image Compression", Doctorate hesis, Beijing University of Posts and Telecommunications.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Genetic Algorithm-like Expression of Reproductive and Competitive RBF network

K. OKUHARA† and T. TANAKA†

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

Abstract

This paper shows the method to transform our neural network to the genetic algorithm-like expression. Proposed neural network, the reproductive and competitive radial basis function network, has also selection by competition, reproduction by bifurcation process, purtabation by the stochastic noise. The result say that we can try considering the reproductive and competitive radial basis function network as the genetic algorithm-like expression.

1 Introduction

We proposed the learning algorithm which reproduce the necessary neurons effectively. This algorithm is realized by reproducing a new neuron taking over some properties which are acquired during learning. The reproductive and competitive radial basis function network (RC-RBFN) [1] is modeled by adding the proposed algorithm to a competitive radial basis function network (CRBFN) [2].

Recently, the genetic algorithm have been applied to many optimization problems [3]. The genetic algorithm mainly consists of four processes, selection, reproduction, crossover and mutation. One of differences between the neural network and genetic algorithm is that the neural network finds the optimal solution by starting from only one initial state, while the genetic algorithm use more than one gene generated at initial time.

2 Outline of the RC-RBFN

2. 1 Relationship Between Combinatorial Optimization Problem and Function Approximation Problem

RBFN is the neural network approximating a nonliner function $\eta(\mathbf{x})$ by a sum of the radial basis function (RBF) [4]. The normalized gaussian active function is generally used for the RBF. Figure 1 shows the RBFN which consists of M input neurons and 1 output neuron. The ith input vector $\mathbf{x}_i \in R^d$,

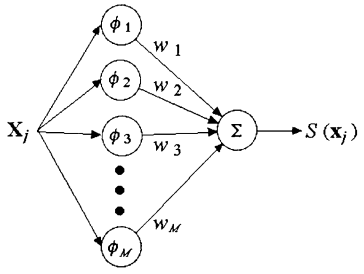


Figure 1 The structure of the RBFN.

$(i = 1, 2, \cdots, N)$ are inputted to all input neurons, where R^d denotes the d-dimensional euclidean spaces. The jth input neuron $(j = 1, 2, \cdots, M)$ has a parameter ϕ_j. The parameter ϕ_j is a set of an mean vector and a covariance matrix $(\mathbf{m}_j, \mathbf{\Sigma}_j)$. $\mathbf{m}_j$ denotes $[m_i^1, m_i^2, \cdots, m_i^d]^{\mathrm{T}}$ and $\mathbf{\Sigma}_j$ $d \times d$ matrix whose inverse matrix $\mathbf{\Sigma}_j^{-1}$ has σ_j^{kl} as the klth component. And $\mathbf{\Sigma}_j$ is positive define symmetric matrix. The jth input neuron outputs

$$\begin{aligned} & \xi(\mathbf{x}_i, \phi_j) \\ = \ & \exp\left\{-\frac{1}{2}(\mathbf{x}_i - \mathbf{m}_j)^{\mathrm{T}} \mathbf{\Sigma}_j^{-1} (\mathbf{x}_i - \mathbf{m}_j)\right\} \end{aligned} \quad (1)$$

for the input vector $\mathbf{x}_i$ where T denotes transpose. Such input neuron is called the RBF. The outputs from input neurons are transmitted to output neuron through the synapse weights, then they are added up and the output neuron outputs

$$s(\mathbf{x}_i, \mathbf{w}, \phi) = \sum_{j=1}^{M} w_j \xi(\mathbf{x}_i, \phi_j) \quad (2)$$

where $\mathbf{w} = [w_1, w_2, \cdots, w_M]^{\mathrm{T}} \in R^M$ and ϕ denotes a set $\{\phi_1, \phi_2, \cdots, \phi_M\}$. The function approximation

by neural network is to represent the non-linear function $\eta(\mathbf{x})$ by the output $s(\mathbf{x}, \mathbf{w}, \phi)$ of network. For the purpose, the function approximation by the RBFN is realized by decreasing the sum of squared error

$$E(\mathbf{w}, \phi) = \frac{1}{2} \sum_{i=1}^{N} E(\mathbf{x}_i, \mathbf{w}, \phi) \tag{3}$$

where

$$E(\mathbf{x}_i, \mathbf{w}, \phi) = \{\eta(\mathbf{x}_i) - s(\mathbf{x}_i, \mathbf{w}, \phi)\}^2 \tag{4}$$

is a squared error function. That is, what the RBFN must acquire in learning are the synapse weight w_j of jth RBF, the parameters $\mathbf{m}_j$ and $\mathbf{\Sigma}_j$.

We have already proposed the survival of the fittest type learning algorithm for the RBFN [5]. The usefull parameters have been defined in the learning algorithm. $\alpha_{jk}(\phi)$ is a competitive coefficient which represents the effect of competition between jth neuron and kth neuron, and it is given by

$$\alpha_{jk}(\phi) = \sum_{i=1}^{N} \xi(\mathbf{x}_i, \phi_j) \xi(\mathbf{x}_i, \phi_k). \tag{5}$$

$\beta_j(\phi)$ is an internal natural increase rate and it is defined by

$$\beta_j(\phi) = \sum_{i=1}^{N} \eta(\mathbf{x}_i) \xi(\mathbf{x}_i, \phi_j). \tag{6}$$

Then the sum of squared error $E(\mathbf{w}, \phi)$ for the function approximation problems can be rewritten by

$$\begin{aligned} & E(\mathbf{w}, \phi) \\ = & \frac{1}{2} \sum_{i=1}^{N} \left[\eta(\mathbf{x}_i)^2 - 2\eta(\mathbf{x}_i) \sum_{j=1}^{M} w_j \xi(\mathbf{x}_i, \phi_j) \right. \\ & \left. + \left\{ \sum_{j=1}^{M} w_j \xi(\mathbf{x}_i, \phi_j) \right\} \left\{ \sum_{k=1}^{M} w_k \xi(\mathbf{x}_i, \phi_k) \right\} \right] \\ = & \frac{1}{2} \sum_{j=1}^{M} \sum_{k=1}^{M} \alpha_{jk}(\phi) w_j w_k - \sum_{j=1}^{M} \beta_j(\phi) w_j \\ & + \frac{1}{2} \sum_{i=1}^{N} \eta(\mathbf{x}_i)^2. \end{aligned} \tag{7}$$

Because the last term is a constant which does not dependent with the parameter $\{\mathbf{w}, \phi\}$, we can define objective function of the function approximation problems as follows:

$$\begin{aligned} & S(\mathbf{w}, \phi) \\ = & \frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{M} \alpha_{ij}(\phi) w_i w_j - \sum_{i=1}^{M} \beta_i(\phi) w_i. \end{aligned} \tag{8}$$

Generally, the objective function of the traveling salesman problem is given by

$$F(\mathbf{v}) = -\frac{1}{2} \sum_{i=1}^{M} \sum_{j=1}^{M} T_{ij} v_i v_j - \sum_{i=1}^{M} I_i v_i \tag{9}$$

where the condition for $\mathbf{v} = [\ v_1,\ v_2,\ \cdots,\ v_M\ ]^{\mathrm{T}}$ is not considered. We can easily show that the objective function of the function apploximation problem is equivalant to one of combinatorial optimization problems by using the following transformation with respect to the coefficients

$$\left\{ \begin{array}{rcl} \alpha_{ii}(\phi) & = & -T_{ii}, \\ \alpha_{ij}(\phi) & = & \alpha_{ji}(\phi) = -\dfrac{T_{ij} + T_{ji}}{2}, \\ \beta_i(\phi) & = & I_i. \end{array} \right. \tag{10}$$

2. 2 Dynamics of RC-RBFN

In the CRBFN, the learning algorithms of parameters $\mathbf{m}_j$ and $\mathbf{\Sigma}_j$ are the same to the usual RBFN, and they are given by the steepest descent method. For the learning algorithm of the synapse weight w_j, the survival of the fittest type learning rule

$$\begin{aligned} \Delta w_j &= -w_j \frac{\partial E(\mathbf{w}, \phi)}{\partial w_j} \\ &= -\epsilon\ w_j \left(\sum_{k=1}^{M} \alpha_{jk}(\phi) w_k - \beta_j(\phi) \right) \end{aligned} \tag{11}$$

is applied, where ϵ is a positive constant.

Then it is shown that the sum of squared error $E(\mathbf{w}, \phi)$ is a Lyapunov function for Eq. (11) as follows:

$$\begin{aligned} & \frac{dE(\mathbf{w}, \phi)}{dt} = \sum_{j=1}^{M} \frac{\partial E(\mathbf{w}, \phi)}{\partial w_j} \frac{dw_j}{dt} \\ = & -\epsilon \sum_{j=1}^{M} w_j \left(\sum_{h=1}^{M} \alpha_{jh}(\phi) w_h - \beta_j(\phi) \right)^2 \\ \leq & \ 0. \end{aligned} \tag{12}$$

We derive the joint probability density function $p(\mathbf{w}, \phi)$ of the parameters of the mean vectors, covariance matrices and the synapse weights of the CRBFN at time when its learning will be finished. We first transform the synapse weight w_j by

$$y_j^2 = w_j. \tag{13}$$

y_j takes arbitrary real number. Then Eq. (11) can be rewritten by

$$\frac{dy_j}{dt} = -y_j \left(\sum_{h=1}^{M} \alpha_{jh}(\phi) y_h^2 - \beta_j(\phi) \right) \equiv f_j(\mathbf{y}) \tag{14}$$

where $\epsilon/2$ is omitted. Eq. (14) satisfies the integral condition

$$\frac{\partial f_j(\mathbf{y})}{\partial y_k} = \frac{\partial f_k(\mathbf{y})}{\partial y_j}. \tag{15}$$

Therefore we can define the potential $V(\mathbf{y}, \phi)$

$$\begin{aligned} V(\mathbf{y}, \phi) &= -\sum_{j=1}^{M} \int_a^{\mathbf{y}} f_j(\mathbf{y}') d\mathbf{y}' \\ &= \frac{1}{2} \sum_{j=1}^{M} \sum_{k=1}^{M} \alpha_{jk}(\phi) y_j^2 y_k^2 - \sum_{j=1}^{M} \beta_j(\phi) y_j^2 \\ &= S(\mathbf{w}, \phi). \end{aligned} \tag{16}$$

Now that, the dynamics of y_j can be derived from the sum of squared error

$$\frac{dy_j}{dt} = -\frac{\partial E(\mathbf{y}^2, \phi)}{\partial y_j}. \tag{17}$$

y_j which is obeying Eq. (14) is updated to the direction of the steepest descent about the sum of squared error $E(\mathbf{y}^2, \phi)$. The results say that if once the dynamics converge to the local minimum solution, it can not avoid from the fixed point. In order to avoid the local minimum solution, we can give

$$\begin{aligned} y_j(t + \Delta t) &= y_j(t) - \frac{\partial E(\mathbf{y}^2, \phi)}{\partial y_j} \Delta t \\ &\quad + \sqrt{Q \Delta t}\, n_j(t) \end{aligned} \tag{18}$$

as the update rule which includes the noise and discretizes Eq. (17), where $n_j(t)$ is the independent random variable obeying the normal density function $N(0, 1)$ with the mean 0and the variance 1. Q is an arbitrary positive constant. Then the joint probability density function $p_\beta(\mathbf{y}, \phi)$ of the mean vectors, the covariance matrices and the synapse weights of the CRBFN satisfy

$$p_\beta(\mathbf{y}, \phi) = Z_\beta^{-1} \exp\left\{-\beta E(\mathbf{y}^2, \phi)\right\} \tag{19}$$

at time when learning will be finished where $\beta = 2/Q$[6]. Z_β is a total partition function and it is defined by

$$Z_\beta = \int_{\mathbf{y}} \int_{\phi} \exp\left\{-\beta E(\mathbf{y}^2, \phi)\right\} d\mathbf{y} d\phi. \tag{20}$$

The minimization about the sum of squared error is equivalent to the minimization about the squared error function $E(\mathbf{x}_i, \mathbf{w}, \phi)$ for each input vector $\mathbf{x}_i$. Then we consider the mean vector $\mathbf{m}_{j[i]}$ depending on each input vector $\mathbf{x}_i$, and pay attention to the jth RBF at the convergence of learning. The conditional probability density function of the input vector $\mathbf{x}_i$ can be derived by

$$\begin{aligned} p_{\beta'}(\mathbf{x}_i | \mathbf{m}_{j[i]}, \phi'_j, \phi''_j) &= Z_{\beta'}^{-1}(\mathbf{m}_j, \phi'_j, \phi''_j) \\ &\times \exp\left\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]}, \phi'_j, \phi''_j)\right\} \end{aligned} \tag{21}$$

where the parameter ϕ'_j is a set of the synapse weights w_j and the covariance matrices $\mathbf{\Sigma}_j$ about the jth RBF, the parameter ϕ''_j is a set of the synapse weights, covariance matrices and mean vectors which excepts one of the jth RBF. To simplify the followed description, the parameters ϕ'_j and ϕ''_j are simplified. The total partition function is defined by

$$Z_{\beta'}(\mathbf{m}_j) = \sum_{i=1}^{N} \exp\{-\beta' E(\mathbf{x}_i, \mathbf{m}_{j[i]})\}. \tag{22}$$

Let the free energy be

$$F_{\beta'}(\mathbf{m}_j) = -\frac{1}{\beta'} \log Z_{\beta'}(\mathbf{m}_j). \tag{23}$$

In RC-RBFN, we give the update rule of the parameter m_j^k as follows:

$$\begin{aligned} \Delta_{\beta'} m_j^k &= -\epsilon \sum_{i=1}^{N} \frac{\partial F_{\beta'}(\mathbf{m}_j)}{\partial m_{j[i]}^k} \\ &= -\epsilon \sum_{i=1}^{N} \frac{\partial F_{\beta'}(\mathbf{m}_j)}{\partial Z_{\beta'}(\mathbf{m}_j)} \frac{\partial Z_{\beta'}(\mathbf{m}_j)}{\partial m_{j[i]}^k} \\ &= \sum_{i=1}^{N} p_{\beta'}(\mathbf{x}_i | \mathbf{m}_{j[i]}) \Delta m_{j[i]}^k \\ &= < \Delta m_j^k >_{\beta'} \end{aligned} \tag{24}$$

where m_j^k is the kth component of the parameter $\mathbf{m}_j$ and

$$\Delta m_{j[i]}^k = -\epsilon \frac{\partial E(\mathbf{x}_i, \mathbf{m}_{j[i]})}{\partial m_{j[i]}^k}. \tag{25}$$

We describe the differences of learning algorithm between the usual RBFN and the RC-RBFN. The learning algorithm, $\Delta w_j \equiv dw_j/dt$, $\Delta m_j^k \equiv dm_j^k/dt$, $\Delta \sigma_j^{kl} \equiv d\sigma_j^{kl}/dt$, of the RBFN is generally derived by applying the Delta rule [7] to Eq. (7), on the other hand the update rules of the RC-RBFN are given by Eq. (11), Eq. (24) and

$$\Delta \sigma_j^{kl} = -\epsilon \frac{\partial E(\mathbf{w}, \phi)}{\partial \sigma_j^{kl}}. \tag{26}$$

Compared with the usual RBFN, the RC-RBFN can can avoid the delay and overfitting of learning.

3 Proposal of Genetic Algorithm-like Expression for the RC-RBFN

We generate the gene j $(j = 1, 2, \cdots, M)$ for the synapse weight w_j and the average m_j^k, as follows:

$$w_j \rightarrow < b_0^j\ b_1^j\ b_2^j\ \cdots\ b_{N_w}^j >, \tag{27}$$

$$m_j^k \rightarrow < c_0^{jk}\ c_1^{jk}\ c_2^{jk}\ \cdots\ c_{N_m}^{jk} > \tag{28}$$

where b_i^j and c_i^{jk} take 0 or 1. They satify

$$w_j = w_{min} + g(w)\left(\sum_{i=0}^{N_w} 2^i b_i^j\right), \tag{29}$$

$$m_j^k = m_{min} + g(m)\left(\sum_{i=0}^{N_m} 2^i c_i^{jk}\right) \tag{30}$$

where

$$g(x) = \frac{x_{max} - x_{min}}{2^{N_x+1} - 1}. \tag{31}$$

The ith string b_i^j relating to the synapse weight w_j changes its state by

$$\begin{cases} u_i^{(n+1)} = r u_i^{(n)} + (1-r)\sqrt{u_i^{(n)}}\, b_i^{j(n)} \\ b_i^{j(n)} = f\left(-2^i g(w) \nabla E_{u_i^{(n)}}\right) \end{cases} \tag{32}$$

where r takes arbitrary number from 0 to 1 and

$$\begin{aligned} \nabla E_{u_i^{(n)}} &= \left.\frac{\partial E(\mathbf{u}, \phi)}{\partial u_i^{(n)}}\right|_{\mathbf{u}=\mathbf{u}^{(n)}} \\ &= \sum_{j=1}^{M} \alpha_{ij}(\phi) u_j^{(n)} - \beta_i(\phi). \end{aligned} \tag{33}$$

Because of $b_i^j \in [0, 1]$, we assume

$$f(x) = \begin{cases} 0 & (x < 0), \\ 1 & (x \geq 0). \end{cases} \tag{34}$$

We suppose

$$m_0^k < m_1^k < m_2^k < \cdots < m_M^k. \tag{35}$$

The crossover between m_p^k $(p = 0, 1, \cdots, M-1)$ and m_q^k $(q = p+1)$ should be done when they satisfy

$$\Delta_{\beta'} m_p^k > 0 \ \ and \ \ \Delta_{\beta'} m_q^k < 0 \tag{36}$$

$$or \quad \Delta_{\beta'} m_p^k < 0 \ \ and \ \ \Delta_{\beta'} m_q^k > 0. \tag{37}$$

In order to detect $\Delta_{\beta'} m_j^k = 0$, we execute crossover at point i which satisfies

$$c_i^{jk} = 1 - f(\Delta_{\beta'} m_j^k), \qquad (j = p, q). \tag{38}$$

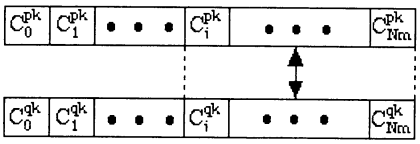


Figure 2 The conceptual figure of crossover.

4 Summary

In this paper, we show the method to transform our neural network to the genetic algorithm-like expression. Proposed neural network, the reproductive and competitive radial basis function network, has also selection by competition, reproduction by bifurcation process, purtabation by the stochastic noise. The result say that we can try considering the reproductive and competitive radial basis function network as the genetic algorithm-like expression.

References

[1] K. Okuhara, and S. Osaki, "Competitive Radial Basis Function Networks Applying the "Survival of the Fittest Type" Learning Algorithm," *IEICE Trans.*, vol. J80-D-II, no. 12, pp. 3191-3199, 1997.

[2] K. Okuhara, and S. Osaki, "Competitive Radial Basis Function Networks Applying the "Survival of the Fittest Type" Learning Algorithm," *IEICE Trans.*, vol. J80-D-II, no. 12, pp. 3191-3199, 1997.

[3] D. E. Goldberg, "*Genetic Algorithms in Search, Optimization, and Machine Learning*," Addison Wesley, 1989.

[4] J. Park, and I. W. Sandberg, "Universal approximation using radial basis function networks," *Neural Computation*, **3**, pp. 246-257, 1991.

[5] K. Okuhara, and S. Osaki, "Analysis of the Synapse Plasticity Equation Considering the Dale Rule," *Trans. ISCIE*, **8**, No. 12, pp. 718-720, 1995.

[6] K. Okuhara, and S. Osaki, "Formation of a Potential of Simplex Systems by the Identification of Parameters," *Technical Report of IEICE*, NC97-15, June, 1997.

[7] B. Widrow, and R. Winter, "Neural nets for adaptive filtering and adaptive pattern recognition," *IEEE Computer*, **21**, pp. 25-39, 1988.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Introducing a New Genetic Operator, Schema Extraction, to Solve the Traveling Salesman Problem

Yong-Gun Jo and Hoon Kang

School of Electrical & Electronics Engineering,
College of Engineering, Chung-Ang University,
221 HeukSuk-Dong, DongJak-Gu, Seoul 156-756, Korea
E-mail : infinity@sirius.cie.cau.ac.kr, hkang@cau.ac.kr

Abstract

In this paper, we introduce a new genetic reordering operator based on the concept of schema to solve the Traveling Salesman Problem(TSP). Because TSP is a well-known combinatorial optimization problem and belongs to a NP-complete problem, there is a huge solution space searched. For robustness to local minima, the operator separates selected strings into two parts to reduce the destructive probability of good building blocks. And it applies inversion to the schema part to prevent the premature convergence. At the same time, it searches new spaces of solutions. In addition, we have the non-schema part to be applied to inversion as well as for robustness to local minima. Comparing with traditional reordering operator, the new operator has good and reliable results through simulation results.

Keywords: TSP, genetic algorithm, schema theorem

1 Introduction

Given N cities and distances among them, the goal of the Traveling Salesman Problem (TSP) is to find an ordering of cities that makes the shortest tour for a virtual salesman to travel each city only once and to return to the city from which he started. TSP is a representative combinatorial optimization problem and belongs to a class of problems known as NP-complete"Garey,Johnson[1]". It minimizes the tour length a salesman travels but the number of possible tours existing in the search space of solutions increases exponentially as one city is added. The various researches have been carried out to find near optimal solution. Simulated annealing"Kirkpatrick et al[2]" executes iterative global search to solve TSP according to the schedule of the temperature transition and cooling rates. The Hopfield network"Hopfield, Tank"[3] defines neurons as cities and minimizes the energy function inducted from the constraints of TSP by recall process. Genetic Algorithms (GAs) search the space of solutions by producing more offsprings whose parents have higher fitness values, based on the mechanics of natural selection and natural genetics"Holland[4] Goldberg[5]".

GAs, first invented by John Holland, mimic the mechanism of biological evolution. They select good strings of chromosomes, i.e. ones with higher fitness to the environment or given problem, then produce better offsprings by genetic operators: reproduction, crossover, and mutation. Reproduction has good parent strings bear more children than others, crossover mixtures the good parts of parents to reproduce fitter offsprings than former generations. Mutation protects from losing some potentially useful genetic material, so that it provides possibility to find the optimal solution and the variety of population preserved and improved.

This paper is supported by Brain Science Research Center of Korean Ministry of Science and Technology under the grant no. 98-J04-01-01-A-07

Each strings in population is regarded as one search point in the search space. As generation goes, the initial population evolve in the way to contain solutions close to optimum. Although GAs have simple structures, they show the powerful performance over the other algorithms. Holland explained this through schema theorem: short, low-order, above-average schemata receive exponentially increasing trials in subsequent generations. Until recently this theoretical foundation was the basis of almost all subsequent theoretical work on genetic algorithms.

In general TSP with GAs, the order of visiting cities is represented as chromosome and an allele's meaning is position-independent. Therefore, crossover and mutation, the basic operators of the simple GA, cannot be used because they will destroy the important orderings with high probability. For this reason, a lot of new GA operators were introduced: inversion, partially matched crossover(PMX), order crossover(OX), cycle crossover(CX), etc. We call them reordering operators in general. Their performances vary along their own probabilities. Inversion shows the best performance among them but still can not guarantee the global minimum.

In this paper, we introduce a new genetic reordering operator, Schema Extraction(SE). It separates selected strings into two parts, schema and non-schema to reduce the destructive probability of good building blocks. And it applies inversion to the schema part to prevent the premature convergence. At the same time, it searches new spaces of solutions. In addition to, we have the non-schema part to be applied to inversion as well as for robustness to local minima. We simulated it to compare with inversion and PMX and show its performance.

2 Schema Theorem

GAs are simple for us to understand conceptually and can be implemented by computer programs for desired solutions easily. However, their behavior can be complicated and there are many considerations for applying GAs to real problems. They include coding methods from which the right strings will be represented, fitness evaluation methods, construction methods of initial population, and the proper probabilities of crossover and mutation. Nevertheless, GAs work great by discovering, emphasizing, and recombining good building blocks of solutions in a highly parallel fashion. This is known as the Building Block Hypothesis"Goldberg[5]". Holland introduced the notion of schemata to formalize building blocks. A schema is a set of bit strings that can be described by a template made up of alleles and asterisks. Suggesting the Schema Theorem, he showed that short, low-order schemata whose average fitness

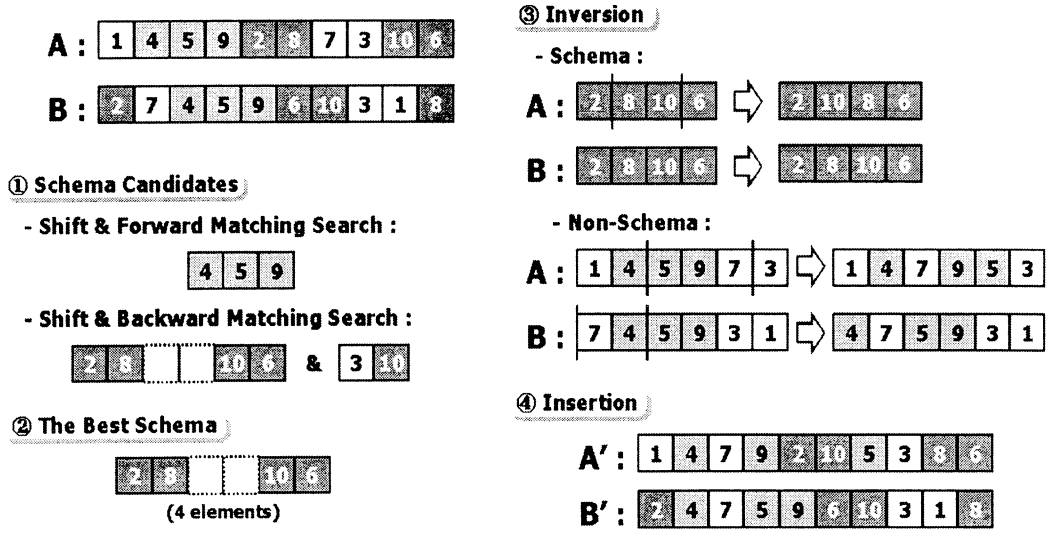


Fig. 1 A schematic diagram of Schema Extraction

remains above the mean of population will receive exponentially increasing numbers of their instances over generation, since the number of instances of those schemata that are not disrupted and remain above average in fitness increases by a factor of $f(H)/\bar{f}$ (i.e., the ratio of the average fitness of the schema to the average fitness of the population) at each generation.

3 Schema Extraction

In GAs, We should use special genetic operators so called reordering operators to solve TSP since an allele's meaning is position-independent. The representative reordering operators are inversion, PMX, OX, CX. Among these operators, inversion has the best performance in time and the optimal solution. PMX and CX appear to have premature convergence problem. OX shows impressive results at some probabilities but its errors are greater than inversion.

We suggest Schema Extraction operator which extracts a set of matching genes from two selected combinatorial strings. The selected strings are possible to have good results so that we can consider the extracted set as schema candidates. Among these ones, the operator chooses the best one and separates it from strings. And inversion operator is applied to it and the rest ones, then the outputs are reinserted into the original positions. Fig.1 depicts the schematic diagram of SE and The procedure of SE is shown as follows.

```
procedure SchemaExtraction()
begin
  do
    shift(p1 & p2)
    forward_matching(p1, p2)
    backward_matching(p1, p2)
    update_number(schema)
    update_distance(schema)
  while (not end of chromosomes)
  compare number of schema with
          the max. number
  update_max_number()
  compare distance of schema with
          the min. distance
  update_min_distance()
  reinsert(inversion(schema) &
          inversion(non-schema))
end
```

Since each element is independent of its position, it is hard to obtain the best schema just by comparing two strings. All the possible cases must be considered. So we put shift, forward-matching and backward-matching process into the algorithm. After this stage, we can collect schema candidates of two strings and elect the best schema among them according to the conditions as follows.

① the number of defined bits is large.

② the fitness of schema is high.

The condition 1 has priority over the condition 2. As shown in Fig.1, inversion is applied to both schema and non-schema as the secondary operator. Inversion for schema part provides the opportunity to jump out of local minimum. On the other hand, inversion for non-schema gives chances to obtain good allele orderings that will subsequently permit more efficient propagation of building blocks. By doing so, SE acquires robustness for optimal solution.

4 Simulation Results

Three probabilities are required for SE: probability of applying SE to the selected strings, of inversion for schema, and of inversion for non-schema. For the 1st probability, we restricted its value under heuristic number of 30% because of preventing the average fitness of population from remaining steady-state or decreasing. We simulated TSP solving by SE and inversion with 1000 population over 1000 generation.

Fig. 2. and Fig. 3. shows the results for inversion and SE, respectively. The enormous experiments revealed that the proper probabilities of SE for 25-city TSP was (30%,50%,50%). In the case of inversion, 20% made best results. The bottom picture of Fig. 3 shows SE escaped from local minimum around generation 850. Fig. 4. represents the fitness comparison between SE(30%,50%,50%) and inversion(20%). This results

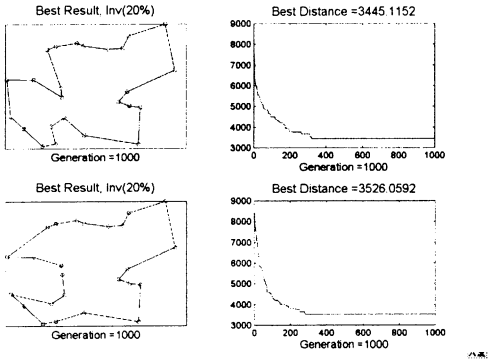


Fig. 2. 25-city TSP solved by inversion

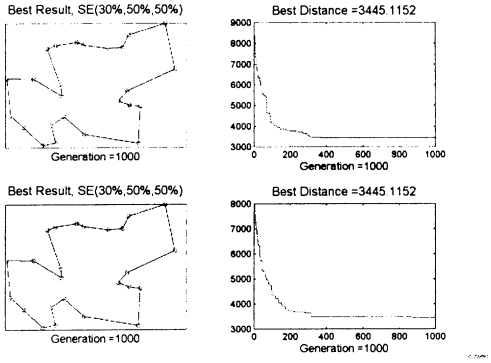


Fig. 3. 25-city TSP solved by SE

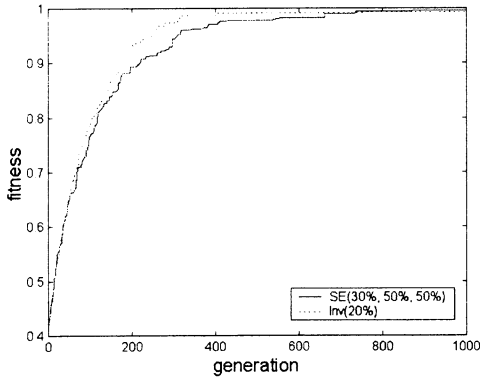


Fig. 4. The fitness comparison between SE and inversion

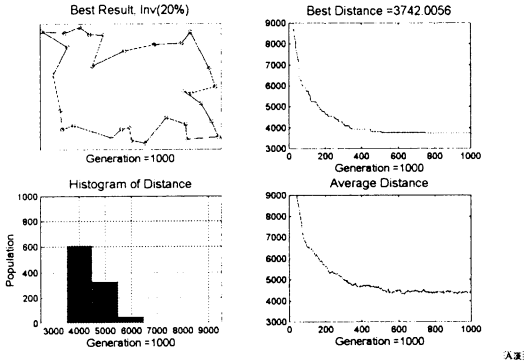


Fig. 5. 30-city TSP solved by inversion

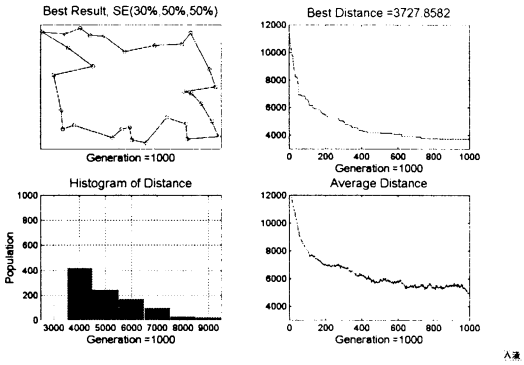


Fig. 6. 30-city TSP solved by SE

obtained by averaging the 5 trials. It shows that SE finds more optimal solution than inversion and possibility of premature convergence of inversion. We proceed 30-city TSP, the results appear in Fig. 5. and 6.

5 Conclusion

In this paper, we suggest a new genetic operator, Schema Extraction, based on the concept of schema. By separating schema and non-schema, more instances of possible building blocks are preserved and reproduced. The simulation results shows SE provides fast initial convergence and robustness to local minima.

References

[1] Garey MR, Johnson DS(1979), Computers and Intractability: a guide to the theory of NP-completeness. Freeman, New York

[2] Kirkpatrick S, Gelatt CD, Vecchi MP(1983), Optimization by simulated annealing, Science 220: 671-680

[3] Hopfield JJ, Tank DW(1985), 'Neural' computation of decisions in optimization problems. Biological Cybernetics, 52:141-152

[4] Holland JH(1975), Adaptation In Natural and Artificial Systems, The University of Michigan Press, Ann Arbor

[5] Goldberg DE(1989), Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, MA

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A Reinforcement Learning Scheme based on Decision Tree Representation of State Space and Its Genetic Acquisition

Hiroshi FUJIKI, Isao ONO, and Norihiko ONO
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 (RL) agents synthesize collective decision policies needed to accomplish their common goal effectively. Most of these straightforward RL approaches, however, scale poorly to more complex multi-agent learning problems, because the state space for each learning agent grows exponentially with the number of its partner agents engaged in the joint task. To remedy the exponentially large state space in multi-agent reinforcement learning (MARL), we propose an approach to MARL which represents the state space for a RL agent using a decision tree, and a genetic programming based framework for identifying an appropriate decision tree representation of the state space. The effectiveness of the proposed idea is demonstrated through the application to a multi-agent learning problem.

1 Introduction

Recent attempts to let monolithic reinforcement learning (RL) agents synthesize coordinated relationships among them scale poorly to more complicated multi-agent (MA) learning problems where multiple learning agents play different roles and work together for the accomplishment of their common goal. These agents have to perform their learning tasks receiving and responding to a variety of sensory information from their partners as well as that from the physical environment itself, and hence their state spaces are subject to grow exponentially with the number of the partners.

To remedy the problem of combinatorial explosion in multi-agent reinforcement learning (MARL), we previously proposed a modular approach [2]. The results obtained based on this approach are encouraging, i.e., modular RL agents are able to synthesize highly coordinated behavior with much less memory and time resources, compared with those needed for monolithic RL agents. However, this approach still has serious limitations.

First, in the modular approach, a RL agent represents its state with a combination of its learning module's states, and each module represents its state with a combination of some specific attribute values included in the agent's perceptual inputs. Since a module do not care all the attributes, its state space is often much smaller than the agent's one. However, the agent can not change its state representation dynamically according to its situations, which makes difficult to reduce its state space much more dramatically. Second, the modular approach also supposes that Q-learning is used as its underlying RL algorithm, and is not guaranteed to be compatible with the other RL algorithms.

To eliminate these limitations, we propose yet another approach to MARL which implements the state space for a RL agent using a decision tree, and a Genetic Programming(GP) based framework for identifying an appropriate decision tree representation of the state spaces for multiple RL agents. The effectiveness of the proposed idea is demonstrated through the application to a multi-agent learning problem called the *Simulated Dodgeball Game* -II($\mathcal{SDG}-II$)[3].

2 Proposed Scheme

2.1 Decision Tree based State Space Representation

In this paper, we propose an approach to MARL which implements the state space for a RL agent, using a decision tree. The decision tree consists of a root, internal and leaf nodes, where some logical expression is associated with its root and internal nodes. A RL agent uses its corresponding decision tree as a state

classifier as follows: (i) upon receiving a perceptual input, the agent tests the logical expression associated with the root, (ii) if the result is true, the agent follows the left edge outgoing from the root; otherwise, it follows the right one, and (iii) if the node encountered next is a leaf, the agent recognizes that it is placed at an state corresponding to the leaf; otherwise, the agent tests the expression associated with the newly encountered node, and so on. At the moment, we suppose that the decision tree is binary and that a logical expression is limited to an equality test between an attribute included in the perceptual inputs and a constant value.

This approach has two potential advantages over the modular approach. First, it is possible for a R-L agent to dynamically select its state representation, i.e., a path from the root to a leaf of its corresponding decision tree, from a repertoire, i.e., set of all such paths, which should make the agent's state space extremely small. Second, the approach does not depend on any specific RL algorithms and it can be used in conjunction with most RL algorithms. The problem is that it is impossible for a human designer to identify an appropriate decision tree representation of the state space for a RL agent. Thus, we have to devise a framework for identifying such representation automatically.

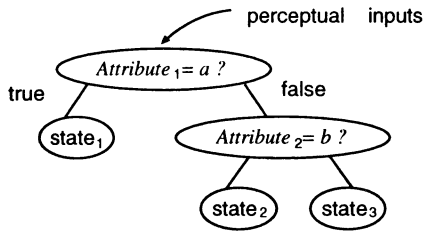


Figure 1: Decision tree representation of state space. In this case, all the states are classified into the same one($state_2$) when an agent receives perceptual inputs $< attribute_1 = a, attribute_2 = b, \cdots >$.

2.2 GP-based Acquisition of State Space Representation

We propose a Genetic Programming(GP)[1] based framework for identifying an appropriate decision tree representation of the state space for multiple homogeneous RL agents. This framework evolves a population of individuals, i.e., decision trees, each representing the agent's homogeneous state space. The fitness of a decision tree is evaluated by the performance of those agents under given computational resources, each employing the tree as their common state space representation.

Operators: We do not need any individual(decision tree) with redundant structures. Any path on a decision tree must not have two nodes associated with the same test. To satisfy this condition, an operator called **NORMALIZE**[7] is applied to a newly generated individual. As sown in Fig. 2, this operator modifies an individual, if and only if it violates the condition, i.e., two nodes on a path from the root to a leaf of the individual are associated with the same test.

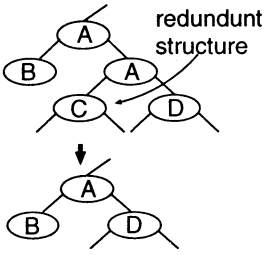


Figure 2: NORMALIZE operator

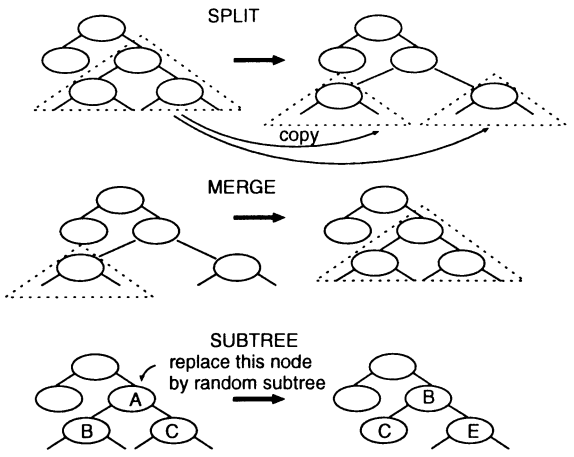


Figure 3: Mutations

We employ the subtree exchange operator. We also employ the following mutation operators:

SPLIT: Choose a subtree (let it be t), and replace t by a newly generated tree which consists of a root associated with a randomly created test and two subtrees, each a copy of t.

MERGE: Replace a randomly chosen subtree by its left or right subtree, each chosen with probability 0.5.

SUBTREE: Replace a randomly chosen subtree by a subtree generated randomly.

As mentioned above, the operator **NORMALIZE** is applied each time when an individual undergoes these genetic operators. Additionally, any subtrees of an individual is eliminated if they have not been accessed throughout its fitness evaluation process.

Generation alternation model: We update the population using a variant of the *Minimal Generation Gap*(MGG[4]) model. It updates the population as follows: (i) It randomly selects two parents from the population, (ii) creates N_c children of them via the crossover with probability p_c, (iii) applies one of the above-mentioned mutations to the children, each with probability $\frac{p_m}{3}$, (iv) selects the two best individuals from the parents plus children, and (v) replaces the parents by the selected ones. Different from the original MGG model, the variant re-evaluates the parents as well as their children, because the fitness of an individual takes a nondeterministic value.

3 Experiments

3.1 Problem Domain

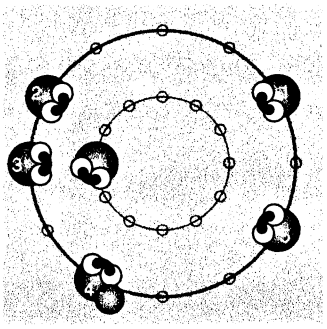


Figure 4: SDG-II

To investigate the effectiveness of the proposed ideas, we applied our RL scheme to a multi-agent learning problem, called the *Simulated Dodgeball Game*-II [3], where multiple RL agents are required to obtain highly coordinated behavior to accomplish their common goal effectively, as follows:

- A single *dodger* agent and multiple *attacker* agents are operating in the discretized game space as shown in Fig. 3.1. The outer circle as well as the inner one is uniformly divided into n_s segments. 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 segments 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 segment, and they can remain at their current segment. 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 segments. A shot succeeds if and only if the shooter and the dodger exactly face each other.

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[1]. 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 coordinated behavior, based on some RL algorithm.

3.2 Implementation

The fitness of an individual is evaluated based on the performance by the corresponding team of attackers. Basically, the performance of a team is evaluated average time steps per successful trial. A trial begins with a single dodger agent and multiple attacker

[1]To reduce the distance, the receiver has to make a step when catching the ball.

agents placed at random segments, succeeds when a shot is made successfully, and it fails when (i) a shot is fails, (ii) a pass is made but it fails, or (iii) the dodger is not hit within n_t time steps. Then, individual agents are relocated at random segments. Since all the trials do not always ends successfully, we evaluate the performance of a team by the average *length* of trial, where the lengh of a successful trial is simply the number of time steps for the attackers to hit the dodger, while the length of a failed trial is n_t.

To evaluate the fitness of a team, (1) we give some prescribed number LT of trials (*learning term*) to the team to allow its members synthesize coordinated behavior using some specific RL algorithm, and then (2) we give some prescribed number ET of trials (*evaluation term*) to the team to let it behave based upon thus synthesized behavior and investigate the behavior. The fitness is an average of the performance exhibited by a team during its evaluation term.

As the underlying RL algorithm, we employ Q-learning [6]. Each attacker selects its actions using ϵ-greedy [5] strategy throughout the learning term, while it selects greedy actions throughout the evaluation term.

We suppose that a perceptual input to an attacker consists of the following attributes: (i) distance to the dodger, (ii) distance to the ball-holding attacker, and (iii) a binary string with a fixed length of $n_s - 1$ indicating which positions are occupied by the other attackers. We also suppose that each attribute is represented by a binary string with fixed length.

We employ the following parameter setteings:

- Parameters for the $\mathcal{SDG}$-II: The number of segments N_{seg} and the number of attakers are set to 8 and 3, respectively.
- Parameters for GP: The population size, the number N_c of crossover operators applied per MGG iteration, and maximum length of a decision tree are set to 100, 2, and 6, respectively. Three different learning terms($LT = 1000, 3000$, and 5000) are considered, while the evaluation term ET is fixed to 500.
- Parameters for Q-learning: The learning rate α, discount factor γ, and the probability ϵ are set to 0.1, 0.9, and 0.1 respectively. Initial Q-values for individual state-action pairs are 0. Immediately
- Rewards for RL attacker agents: After performing an action, the attackers receive the following reinforcement signals: (i) when a shot is made successfully, the shooter and the other attackers immediately receive rewards of 1.0 and 0.5, respectively, (ii) when a shot is made but it fails, the shooter receives a penalty of -0.5, and all the others receives a cost of -0.1, (iii) 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, (iv) the attackers receive a cost of -0.1.

3.3 Results

Figure 5 depicts the GP-based evolutionary processes of the decision tree representation of the state space for homogeneous RL attackers in $\mathcal{SDG}$-II. The above-mentioned three different cases($LT = 1000, 3000$, and 5000) are shown. In Fig. 5 (a), the x-axis indicates the number of MGG iterations in the GP processes, while the y-axis indicates the fitness of the best individuals in the population. Figure 5 (b) shows the preformance that the best individuals exhibited during their evaluation terms. In Fig. 5 (b), the x-axis indicates the number of trials the team of attackers undergoes, while the y-axis indicates the average length of trials. Again, the above three cased are shown. These results suggest that the proposed scheme is able to identify appropriate state space representation for multiple homogeneous Q-learning agents engaged in the joint task, under the available computational resources, i.e., given learning terms.

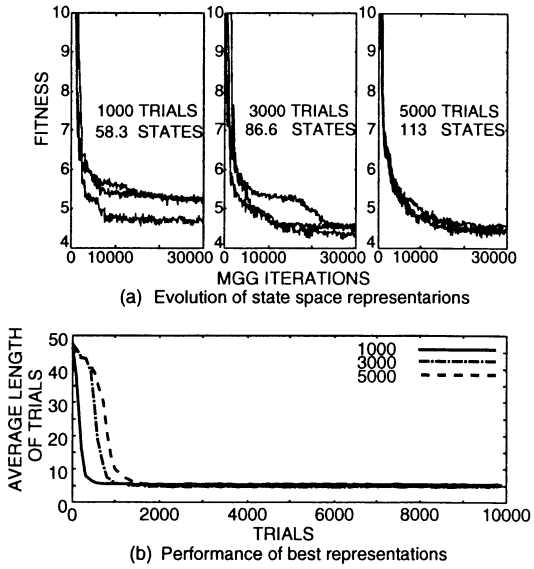


Figure 5: Process of evolution

Table 1 : Proposed scheme vs. Q-learning

Used scheme	*Success rate of trials*	*Average length of trials*
Proposed scheme	0.9992	4.6736
$Q-learning$	0.9998	4.951

To see the efficacy of the proposed scheme compared with a straightforward Q-learning scheme using a single lookup table, we tried to solve the same $\mathcal{SDG}$-II using the straightforward one. Table 1 shows the performance by the best team obtained using the proposed scheme($LT = 3000, ET = 500$) and that of a team consisting of Q-learning attackers($LT = 20000, ET = 500$), each implementing its policy with a lookup table, where $LT = 5000$ and $ET = 500$. The performance shown here is averaged over 10 independent experiments. Note that the best team clearly outperforms the team of standard Q-learning attackers.

4 Concluding Remarks

To remedy the problem of combinatorial explosion in MARL, we proposed a RL scheme based on decision tree representation of an agent's state space. This scheme allows multiple RL agents to perform their learning tasks in extremely small state spaces. We also proposed a GP-based framework aimed at the identification of an appropriate decision tree representation of their state spaces. The results are quite encouraging and suggest that our approach is promising in the study of adaptive behavior of multiple autonomous agents.

References

[1] J. R. Koza:Genetic Programming: On the Programming of Computers by Means of Natural Selection. MIT Press, Cambridge, MA, USA, 1992.

[2] N. Ono, *et al.*: Multi-Agent Learning: A Modular Approach, Proc. ICMAS-96, AAAI Press 1996.

[3] N. Ono, and S. Yoshida: Synthetic Collective Behavior by Multiple Reinforcement Learning Agents in SImulated Dodgeball Game, Proc. of the Fourth International Conference on Artificial Life and Robotics, pp.540-543, 1999.

[4] H. Satoh, M. Yamamura, and S. Kobayashi: Minimal Generation Gap Model for GAs Considering Both Exploration and Exploitation, Proc. 4th Int. Conf. on Soft Computing, pp.494-497, 1996.

[5] Sutton, R.S., and Barto, A.G.: Reinforcement Learning: An Introduction, MIT Press, 1998.

[6] C. J. C. H. Watkins, and P. Dayan: Q-learning. Machine Learning, Vol. 8, No.3, pp. 279-292, 1992.

[7] K. Yoshida: Extension of Decision Tree Induction and Its Engeneering Applications, Doctor thesis, Tokyo Institute of Technology, 1997 (in Japanese).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Approximation of a nonlinear function using neural network based on genetic algorithm

Masanori Sugisaka[1] Fengzhi Dai[2]
Dept. of Electrical and Electronic Engineering
Oita University, Japan
[1] msugi@cc.oita-u.ac.jp [2] daifz@cc.oita-u.ac.jp

Abstract

In this paper, based on the genetic algorithm (GA), we will consider the optimization of the artificial neural network (ANNs) architecture (the type and structure).

The chromosomes generated by GA contain the information of the neural networks, from which can we approximate an unknown nonlinear function.

The multilayer feedforward network is involved in this paper. The weights are obtained by back propagation (BP) algorithm.

Keywords: Artificial Neural Network, Genetic Algorithm, Nonlinear Function Approximation

1 Introduction

In the conventional mathematics analysis field, it is often difficult to obtain the correct result if we want to approximate an unknown nonlinear function that has a large number of input variables [1].

And in the time series prediction field [2], faced to a large number of unceasing changing data, what shall we do?

Also in the system control field, if we want to obtain the model and the parameters of the control plant, especially for the nonlinear analogous system, we often use the model reference adaptive control method (MRAC) or self-regulators [3]. But we should consider the problem of Lyapunov and/or Popov stability. And the model is often not accurate because of all kinds of factors that we have considered and not.

In this paper, based on GA, we will consider the optimization of the ANN's architecture (the type of ANNs, the number of the input and hidden layer neurons, the type of functions of each neuron), from which can we approximate an unknown nonlinear function.

We know that the most important feature of ANNs is their learning ability. ANNs learn from experience, generalize from previous examples to new ones, and abstract characteristics from inputs. Recent advances in ANNs have led to several applications such as speech recognition, image restoration, and nonlinear function approximation.

ANNs have the advantage of organizing all the data altogether to find out their inner relationships. GA is a good method to train/test these data.

In the ANN's design and development project, there are a lot of tasks: splitting data into training and testing sets, determining which input variables to use, selecting the ANN's type and optimizing the structure. Fig.1 gives some factors that will influence the ANN's performance [2].

Fig.1 factors that will influence the ANN's performance
the number of data records
the number of input variables
the number of the hidden layers and the hidden neurons per layer
the size of population
the type of accuracy calculation
the type of ANN that we want to use

The conventional process is at first to select the neural network type, the number of the input and hidden layer neurons, and then train/test data. These selections are mainly based on experience and some special methods [4].

In this paper, we consider the automatic neural network architecture design. That is to say, the chromosomes generated by GA contain the information of the neural networks, and we use the genetic operators to obtain the best neural network type and structure.

This thought emerged from the need to easily and quickly discover the best data elements and neural network architectures.

2 ANNs Optimization Algorithm

2.1 Genetic Encoding Method

Since the chromosomes generated by GA contain the architecture information of the ANNs, the first step is to encode ANNs into binary strings [5]. The maximum number of input neurons N_{imax} and the maximum number of each hidden layer neurons N_{hmax} must be given in advance.

For a neural network, we should determine the proper number of hidden layers, the number of the hidden neurons per layer, and the transfer function of each neuron (the transfer function is an attribute of neurons, not layers, so we try to design multiple transfer functions per layer).

So one chromosome should contain the input area, the ANN type area, the hidden layer area and the output area. Fig.2 gives an example of an encoded chromosome and Fig.3 is the corresponding explanation. N_{net} represents the type of ANNs, N_o is the number of output neurons.

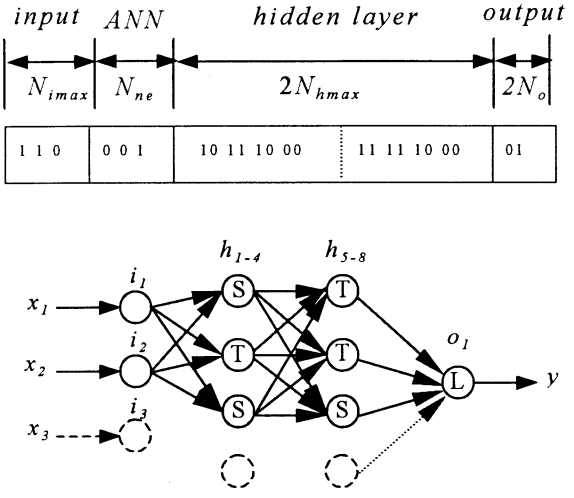


Fig.2 The chromosome and ANN structure

2.2 Training Process Explanation

The initial populations and weights of ANNs are generated randomly. And the ANNs are trained by BP algorithm. BP is a systematic training technique for a multilayer, feedforward ANN, and it is capable of approximating any function [4].

The objective of training is to adjust the weights so that the application of a set of inputs can produce the desired set of outputs.

Area	Binary string	Representation
Input layer	1	existence
	0	not existence
ANN type	001	Multilayer Forward
	010	Time Delay ANN
	...	...
hidden layer and output layer	00	not existence
	01	Linear function (L)
	10	Sigmoid function (S)
	11	Hyperbolic tangent (T)

Fig.3 Genetic Encoding Method

For the function approximating applications, we use random selection method to split the data into train and test data. This method is excellent when we want to avoid possible trending effects that may be inherent in our data, especially if the data was sorted previously.

In this paper, we present an example of how to approximate a nonlinear function (just like a multi-input one-output nonlinear system).

We use a multilayer feedforward ANN to represent this function, and the experiment results are presented.

3 Experiment

3.1 The Training and Testing Data

There are 400 input/output data sets to train and test the ANN to approximate the function:

$$F(x_1, x_2) = x_1^2 + 3x_2 + 1 \qquad (1)$$

For the objective of watching the effect, we add a variable x_3 as the random or noise input variable.

Thus the selected ANN should have 3 inputs and 1 output (Fig.4).

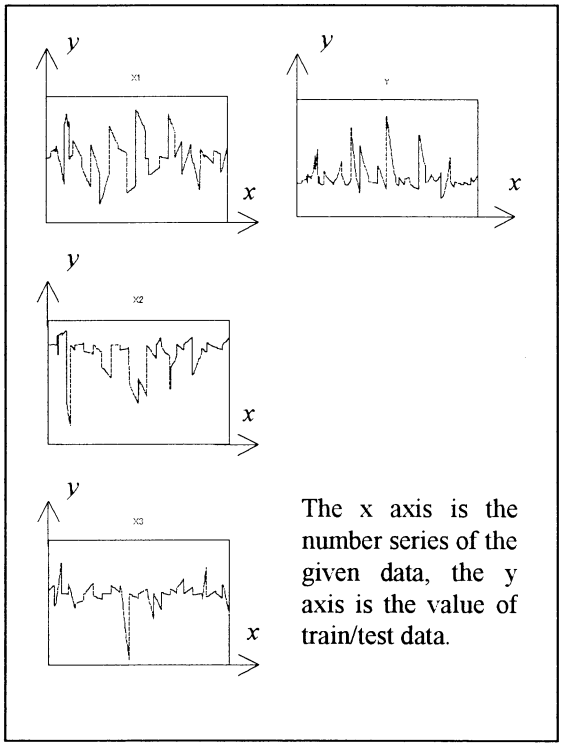


Fig.4 The train/test data

Fig.5 gives some important parameters of the ANNs and GA that are used in this paper.

Records	200 training / 200 testing
Fields	x_1, x_2, x_3, y
Generations	50 (the stopping criteria)
Population Size	30
N_{imax}	3 (x_1, x_2, x_3)
N_{hmax}	8 (in each hidden layer)
Reproduction (selection)	P_r% of the best individuals will be selected to survive, the remained (100-P_r)% will be treated by the genetic operators
Crossover	two-point crossover
Mutation	general random exchange method at the rate of P_m
P_r	50
P_m	0.25

Fig.5 Some ANN/GA parameters

The objective that we use the additional variable x_3 is:

In the system control field, if we want to recognize the control plant, we sampled lots of data from it, some of them are useful, some are useless. Since we do not know which are useful to recognize the plant in advance, we should add all these data as the inputs to train the ANN. This experiment is to watch the effect of those useless data for the recognition of a system.

3.2 Experiment Results

Fig.6 gives the selected best ANN. It was found on the generation 45 after a runtime of 00:42:03. This network employed 2 inputs and 2 hidden layers. Its training average absolute error is AAE_{train} = 5.1463, its testing average absolute error AAE_{test}= 4.8228, which,

$$AAE = \frac{\sum |Y_{actual} - Y_{predicted}|}{n} \qquad (2)$$

In this experiment, the variables x_1 and x_2 are important, x_3 should be useless. But we see that x_3 sometimes is included into the models, we think that it is because there is some false correlation to the predicted output.

As the comparison, if we only use x_1, x_2 as the input variables, the result is:

The best ANN (Fig.7) was found on the generation 20 after a runtime of 00:12:15. This network employed 2 inputs (x_1 and x_2) and 1 hidden layer. The hidden neurons are 1S 1T 1L, and the output neuron uses the Tangent (T) transfer function. Its training average absolute error is AAE_{train} = 7.6811, its testing average absolute error AAE_{test}= 5.0966.

We also have some simulation results based on the method presented above in [5].

From the experiment we can see that this method can automatically deal with the useful and useless data well.

3.3 Fitness Function

The fitness of each fully trained neural network is calculated by the following equation:

$$f = \frac{1}{E}(1 + 0.5(\alpha(1 - \frac{N_i}{N_{i\max}}) + \beta(1 - \frac{N_h}{N_{h\max}}))$$

E is the error of ANNs (equation 2), α and β are coefficients implying the influence of the input neurons and the hidden neurons.

input area	2 (x_1 , x_2)
hidden layer 1	2S 2T 2L
hidden layer 2	3S 1T
output area	1 (1T)

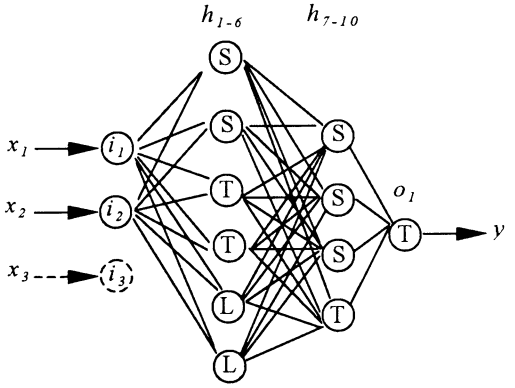


Fig.6 The selected best ANN (3 input variables)

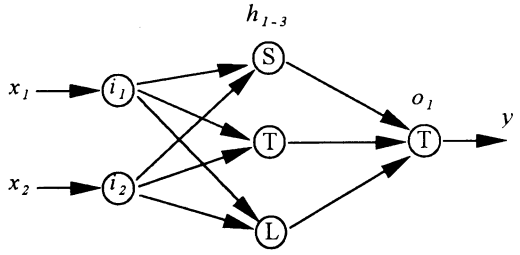


Fig.7 The selected best ANN (2 input variables)

4 Conclusions

The chromosomes generated by GA contain the information of the ANN, and we use the genetic operators to obtain the best artificial neural network. We also use multiple transfer functions to design the ANN.

If we believe that there is a strong functional or mathematical relationship between the data we presented and the results we desire, an unknown nonlinear system then can be represented by a nonlinear function (even this function cannot be written in the pattern of an equation). Then the method presented above can be used to the problem of the identification of a nonlinear system.

The simulation results show that the presented method is suitable in the approximation of a nonlinear function.

Also there are many factors that will impact on the performance of training. For example, since data come from a variety of sources and rarely will it be on a common basis, some initial data preprocessing work will be necessary. And different applications should use different training methods. For example, nonlinear function approximation uses random selection method to split the data into train and test data, while this method is not appropriate for time series prediction, because the sequence of the data records is important to this kind of applications. When we use ANNs, we should consider them.

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 using this method to other applications.

References

[1] Lifan Cao (1986), Numerical Analysis. Beijing Institute of Technology Press, China

[2] The Manual of NeuroGenetic Optimizer (NGO) version2.6. http://www.biocompsystems.com

[3] Dayong Luo (1993), Adaptive control. Railway Press, China

[4] Philip D. Wasserman (1989), Neural Computing: Theory and Practice. Van Nostrand Reinhold

[5] Zhijun Liu and Masanori Sugisaka (1999), Optimization of neural network structures based on genetic algorithm. Proceedings of the 11th SICE Symposium on Decentralized Autonomous Systems, Nagoya, Japan, pp.11-14

[6] Shinya Kitagawa, Keiji Suzuki and Ohuchi Azuma (1999), Incremental generating: Based on GA for Neural Network and Application. Proceedings of the 11th SICE Symposium on Decentralized Autonomous Systems, Nagoya, Japan. pp.205-208

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Neural network-like hierarchical sociogenesis as a common logic underlying evolutionary semeiogenesis and machinogenesis in various levels of organic individuals

Koji Ohnishi and Daiji Kanbe
Dept. of Biology, Fac. of Science, Niigata Univ., Niigata, Ikarashi-2, 950-2181

ABSTRACT: Origin of hierarchical sociogenesis and biomachinogenesis was analyzed from a viewpoint of altruistic and/or co-operative behaviours of bio-individuals which are society-members. Queen-worker type altruistic behavioral networks in bee eusociety and the animal body (= eusociety consisting of germ-line queen-cells and somatic-line worker-cells) were found to show self-learning neural-network-like DNA information flow (including feedback) from workers and queens to the next generation by means of altruism to kin individuals. Possibly similar, but signifi-cantly different neural-network-like DNA-flow (including feed-back) in the intracellular RNA/DNR ribo-organismic society was discussed. Hierarchical sociogenesis and bio-machinogenesis was strongly suggested to have evolved as a phenomenon of cognitive system possessing signifian-signifé-like, and sensory /motor-like elements.

Key words: sociogenesis, machinogenesis, semeiogenesis, kin selection, self-learning neural-network, RNA society, bee eusociety, bee dance

1. Introduction: As has been recently discussed [1], well-made biomachines such as animal body, bee society (= bee super-organism), and intracellular genetic apparatus seem to have most plausibly emerged by hirearchical sociogenesis of lower-level individuals (which are unicell animals, bee individuals, and tRNA ribo-organisms, respectively). Both bee super-organism and animal body are considered to be altruistic kin society consisting of fertile queen individuals (queen bees, germ line unicell organisms) and worker individuals (worker bee, somatic line unicell organisms), where altruistic behavours of the latter to the former are considered to have occurred by the so-called "kin selection" (Hamilton, 1964) [2].

In the emergence of protein-synthesizing and genetic machine, early RNA replicator ribo-organisms would have evolved to be early tRNA ribo-organisms whose life cycle consists of tRNA-phase and tDNA-phase. Such early tRNA individuals would have associated together to make a co-operative or kin tRNA society in which some of them would have altruistically or co-operatively behaved to other tRNAs, and have begun to behave as earliest mRNAs (and also as earliest rRNAs) (as has been discussed in "poly-tRNA theory". See ref.[1].). T-RNAs conserve primitive, ancestral tRNAs, whereas other RNAs (mRNAs, rRNAs, M1 RNA, etc.) have rigourously changed their own structures and functions. Accordingly, it would be rather reasonable to say that mRNA/mDNAs and rRNA/rDNAs (and also M1 RNA and snRNAs) are worker-like tRNA-derived riboorganisms, and contemporary tRNAs are queen-like tRNA riboorganisms. However, the most important difference between bee super-organism (= eusociety) and animal body type hierarchical eusociety and the intracellular hierarchical RNA society is that queen-like tRNAs and worker-like mRNAs are both replicated in their DNA-phase, which means both types of RNAs are fertile. What are the major logical and evolutionary differences between these two types of hierarchical sociogenesis which seems to have made themselves well-made machines ?

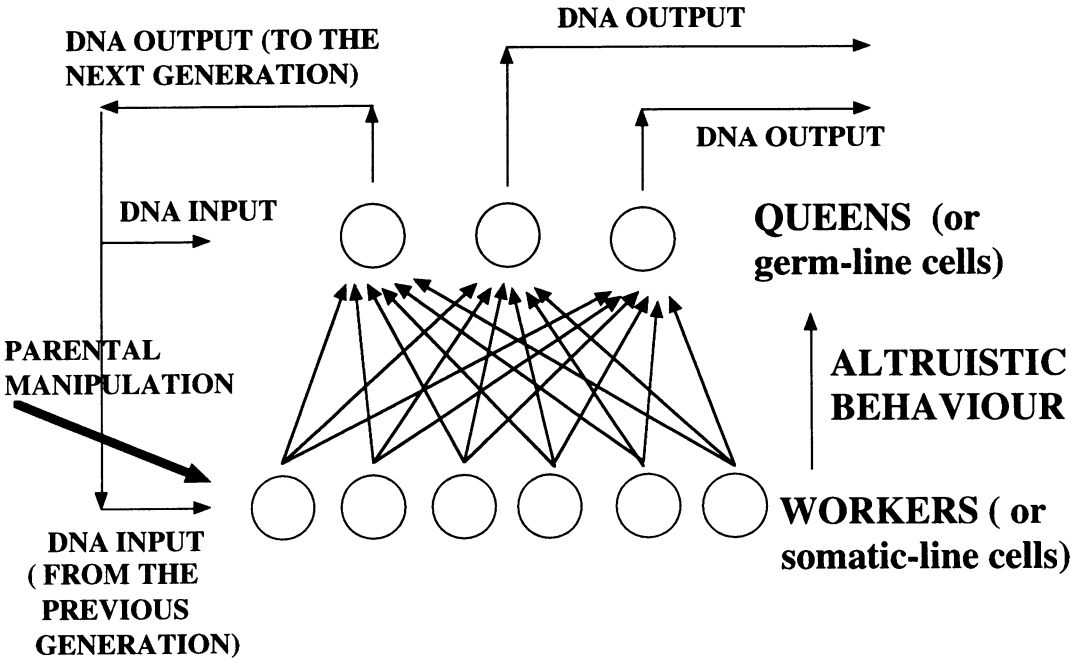
In this paper, this question will be discussed from a viwpoint of the genesis of self-learning neural network by means of hierarchical sociogenesis of altruistic or co-operative behaviours.

2. Neural-network-like DNA-information flow in hierarchical behavioral network of altruistic societies: Why such hiearchical societies could have evolved to be well-made machines ? A possible answer could be that hiearchical altruistic behavioral and DNA-information-flow networks could make a self-learning neural-network machine (Fig.1 [A]). The network would consist of ;

(1) DNA input from the (queen of) previous generation to queens and workers of the present generation,

(2) altruistic behavioral flow from workers to queens, where the behavioral flow is partially equivalent to DNA flow from workers to queens (because altruism increases final DNA-output from queens to the next generation, and the queen's DNAs share a great portion of DNA sequences with workers of this kin society),

[A] Hierarchical Altruistic Behavioral Network of Queen and Worker Kin Individuals (Bee Super-Organism and Multicellular Animal body)



[B] Hierarchical Co-operative Behavioral Network of Queen-like tRNA and worker-like (wl-) RNA riboorganisms (Intracellular Riboorganismic Society)

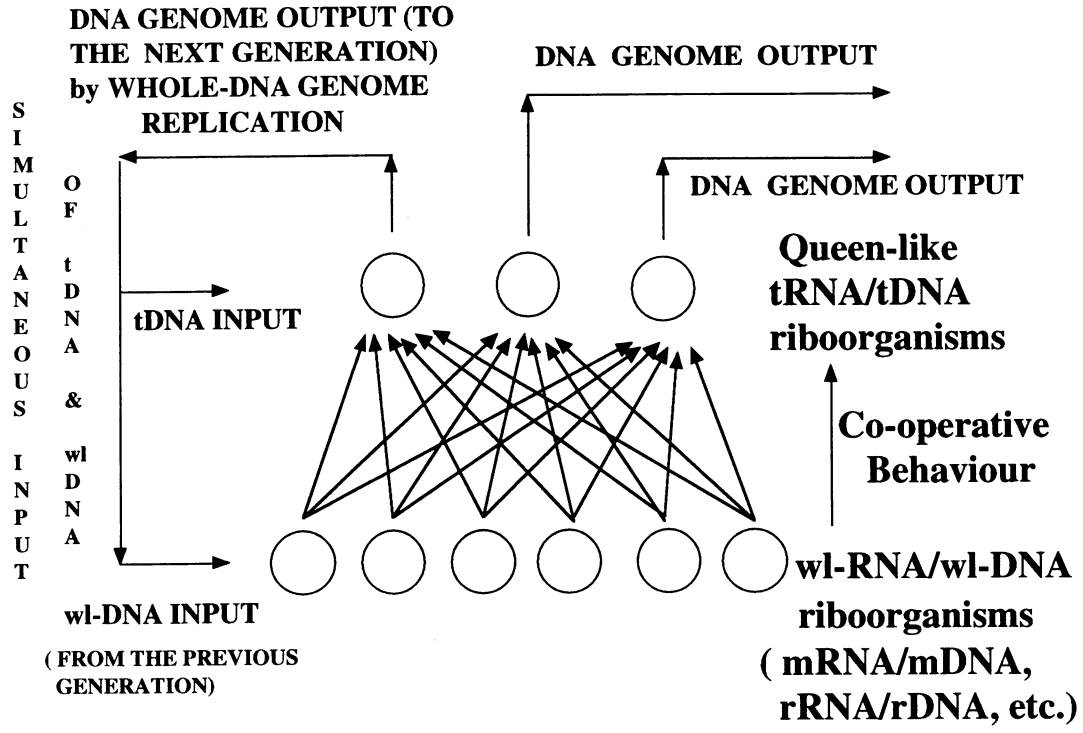


Fig. 1. DNA-information flow in the hierarchical behavioral network in altruistic kin society (bee eusociety and animal body) [A], and in intra-cellular co-operative tRNA/tDNA ribo-organic society.

(3) the final DNA-output from queen to the next generation via "gametes" (i.e., newly grown queen bees in bee-super-organism, or ovum in animal body).

(4) Therefore, the DNA-flows (3) and (1) make a feedback DNA-flow from "queen" (of the previous generation) to "workers" (of the present generation), or else, from "queen-niches" to "worker-niches", and moreover, this feedback might be somewhat like "teacher-infornation" for this possi-ble neural network. Another important feedback would be "manipulation by the previous generation" exemplified by the suppressive effect of queen-bee's (maternal) substance to worker's fertility, and by maternal bicoid-mRNA effect of early cell differentiation in *Drosophila.*

These features (1)-(4) would most plausibly make the kin societies a self-learning neural-network machine capable of self-improving towards evolutionary bio-machinogenesis. In these altruistic societies, the coefficient of (genetic) relatedness (r) between worker and queen is relatively high , the workers can output their own DNA base-sequence informations to the next generation, not by their own DNAs, but by the newly grown fertile females' DNAs (i.e., DNAs of queens of the next generation) via altruism. The quantity of the sum of DNA outputs from all workers to the next generation through the i-th queen is given by,

$$O(Q_i) = \Sigma_j \; r_{ij} \; A(W_j), \qquad \text{[Eq. 1]}$$

where $A(W_j)$ denotes the increase of the i-th (newly grown) queen's DNA-output to the next generation caused by the j-th worker's altruistic behaviour to the i-th queen. r_{ij} denotes the coefficient of genetic relatedness representing the ratio of DNA information quantity shared by the i-th queen and the j-th worker (For example, in case of clonal kin society, $r_{ij} = 1.$). In [Eq. 1], r_{ij} corresponds to the connection weight in neural-netwok theories such as the Back Propaganda method [3], and thereby the DNA informations of the workers are accumulated to queens (fertle females) and further outputted to the next generation via queens. If the DNA informations of workers or queens are more adaptive ones (which tend to increase $A(W_j)$), then such DNA sequences are more effectively transmitted to the next generation via altruism, and would become improved input-informations to the next generation and would improve the bio-machine. Such feed-back system would make the biomachine capable of self-learning .

3. Hierarchical DNA-flow in intracellular RNA society : In case of hiearchical RNA (riboorganismic) society consisting of queen-like tRNAa and worker-like RNAs, the relatedness of riboorganisms is not so strong as can be seen between queen and workers in hymenopteran eusocieties or in clonal kin unicell society making animasl body. Thus, Hamilton's kin selection rule does not seem to have effectively worked throughout evolutionary machinogenesis of genetic apparatus, even if it might have some importance in the earliest emergence of weakly hierarchical structures. The DNA-information flow in the queen-like/ worker-like hierarchical RNA society is schematized in Fig. 1 [B]. In the case of Fig. 1 [A], the DNA flow from worker to the next generation can be achieved by high genetic relatedness and altruistic behaviour. However, in the [B] case, worker-like RNAs (wl-RNAs) also output their own DNA information to the next generation via their own DNA-replication. How the wl-RNAs output their own information to the next generation ? And how does the feedback of DNA informatons occur ? Since wl-RNAs are fertile, hierarchical machinogenesis will not successively evolve unless some appropriate control of worker's behaviours by queen-like tRNAs. tRNA genes (tDNAs) are most plausibly more important and essential than other DNA genes. The feedback control of worker's DNA information seem to be achieved by the mode of DNA replication. The wl-DNAs do not replicate independently but replication of all DNA genes occurs as a replication of the whole genome DNA(s). Such simultaneous replication of all wl- and queen-like DNAs would make efficient evolution of cooperation of different wl-DNAs. In case of independent random replication in intracellular RNA/DNA dsociety, better combination of two or more mutations would soon disappear, and efficient machinogenesis seem to be difficult to evolve. In other words, worker's DNA informations can be directly outputted to ther next generation via replication of DNA genome. Notwithstanding the capability of worke-DNA's replication,
the hiearchy of tRNA ribooorganisms seem to construct some kind of self-learning neural-network machine.

4. Signifian and signifie in general seneiotic systems: In every of these hirarchical societies, mature semeiotic systems are observed ; synaptic signs between sensory and motor neurons, dance-language synapsis between "sensory bee" and "motor bee" [1], triplet codon rules between anticodon (sensory organ or "image" or "signifian" in de Sausseur's terminology) and amino-acid specificity ("concept" or "signifie") [Fig.2].

The bee dance-language system is essentially simlar to the Applasia simple nerve system capable of making habituation and sensitization [(3) and (4) in Fig. 2 . Here, "sensory worker-bee" and "motor-worker-bee" functionally

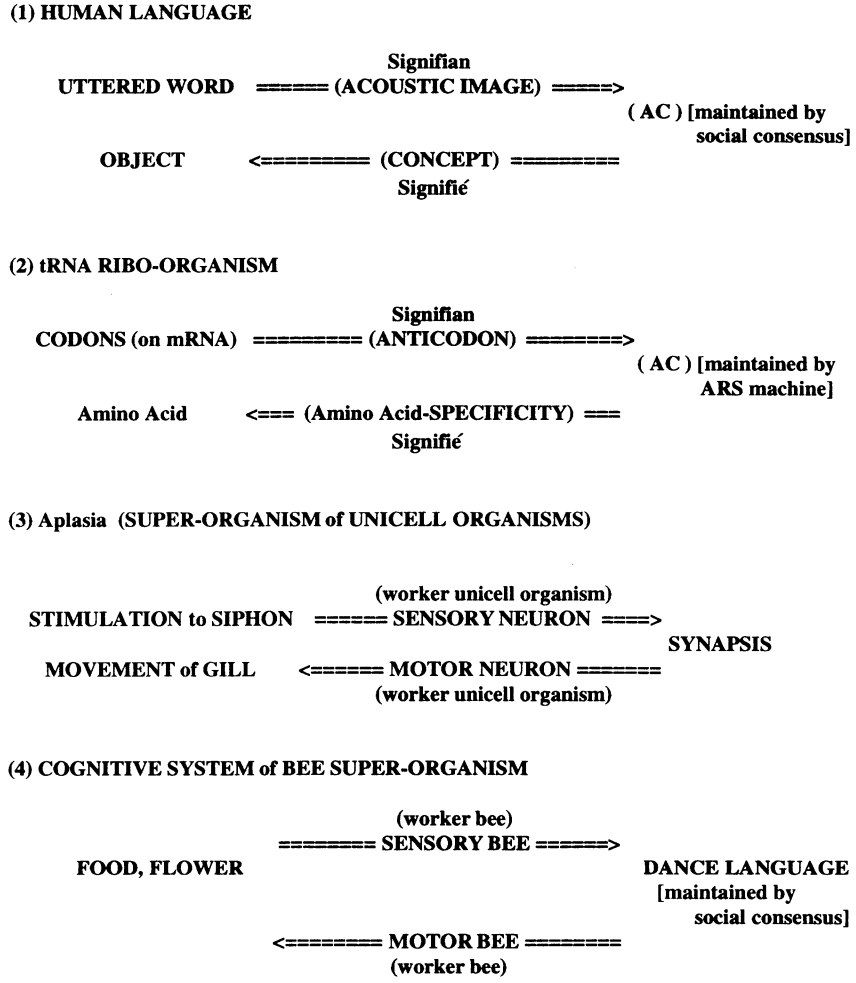


Fig. 2. A schematic representation of signifian-signifié relashinship in general semeiotic systems. (AC): Arbitrary correspondence between signifian and signifié . This correspondence is maintained by the society culture.

correspond to sensory nerve-cell and motor nerve-cell, respectively. Thus, the dance-system and the nerve-cell sestem function as cognitition of bee-suoer-oranism and Aplasia multicellular body, respectively. It is very plausibly concluded that the bee-dance is not a mere communication system among bee individual, but is more plausibly a "sensory system" of the bee-super-organism.

Similarly, the language system of humans would or might have evolved as an adaptive function of human society, rather than a mere comunication system among human individuals. In the case of RNA society, anticodon and amino acid specificity corresponds to D Sausseur's signifian and signifie in the human language system [Compare (1) and (2) in Fig. 2.].

Thus, mature semeiotic systems must have played important roles in every of these machinogenesis. Such semeiotic systems are characteristic cultures of the respective hierarchical societies, which well coincides with de Sausseur's theory that semeiosis is a cultural phenomenon of society. Language system in human society seem to have evolved by a very similar logic. Almost identical logic of semeiogenesis would underlie the evolution of these different semeiotic systems. Genetic codon system is a cultural phenomenon of intracellular tRNA ribo-organismic society, and mRNA-encoded proteins are cultural products (machines) of the society.

References

[1] Ohnishi, K. , S. Hokari, H. Yanagawa: Origin and evolution of early-peptide synthesizing biomachines. Proc. of the 4th Int. Symp. on Artificial Life and Robotics, 344-349 (1999).

[2] Hamilton, W.D.: The evolution of social behaviour. J. Theoret. Biol. 7: 1-52.

[3] Dracopoulos, D.C.: Evolutionary Learning Algorithms for Neural Adaptive Control. Springer, London (1997).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Evolution of Modular Structures for Multiple Reinforcement Learning Agents

Tetsuo Nijo, Isao Ono and Norihiko Ono
Dept. of Information Science and Intelligent Systems
Faculty of Engineering, University of Tokushima
2-1 Minami-Josanjima, Tokushima 770-8506, JAPAN

Abstract

In applying multi-agent reinforcement learning to real world applications, we face a serious problem that the size of the state space grows exponentially with the number of partner agents engaged in the join task. To overcome the problem, the Modular Reinforcement Learning Architecture (MRLA) has been proposed and shown good results. However, the performance of an agent with the MRLA deteriorates unless the agent consists of appropriate modules for a given task, which cannot be predicted in advance. This means that a designer has to identify a good combination of modules by trial and error. In this paper, we propose a framework for automatically acquiring an appropriate combination of modules by evolutionary computation and show its effectiveness by applying it to the Pursuit Problem.

1 Introduction

When applying reinforcement learning to the multi-agent world, we face a serious problem that the state space for a learning agent grows exponentially with the number of its partner agents engaged in the join task.

Some attempts have been made to remedy the exponentially large state space in the multi-agent world. Ono *et al.* applied a variant of Whitehead's modular reinforcement learning architecture [Whitehead 93] to some multi-agent problems as shown in Fig. 1 and demonstrated its effectiveness. The basic concept in the modular approach is that the modular architecture has multiple modules which receive the subsets of the sensory inputs to an agent to reduce the state space.

However, there are no guidelines to design a modular structure, i.e. the number of the modules and the combination of the modules. We should design a modular structure for a given task by trial and error since the modular structure depends on the task.

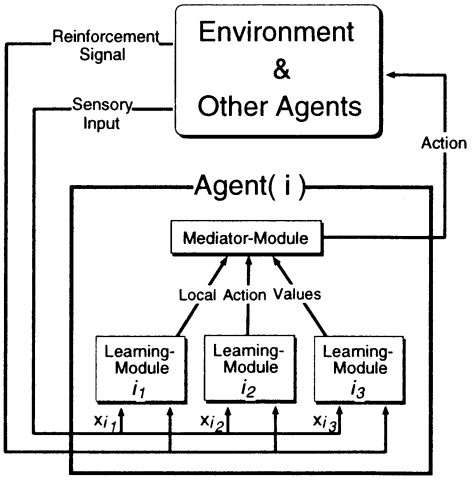


Figure 1: Modular architecture for a agent

In this paper, to overcome the above problem, we propose an evolutionary method for automatically identifying appropriate modular structures of the agents. The method employs a crossover operator which exchanges some modules between two agents and a mutation operator which simply inserts/removes some modules to/from an agent. From a view point of maintaining a diversity in the population, we use a variant of the Minimal Generation Gap (MGG)[Satoh 96] as the generation-alternation model.

To show the effectiveness of the proposed method, we apply it to the Pursuit Problem [Benda 85] and attempt to identify appropriate modular structures for the agents, which are compared with the modular structure designed by a human in [Ono 96].

2 Problem Domain

We consider the modified version of the Pursuit Problem [Benda 85] which was used in [Ono 96]. The brief explanation of the problem is as follows.

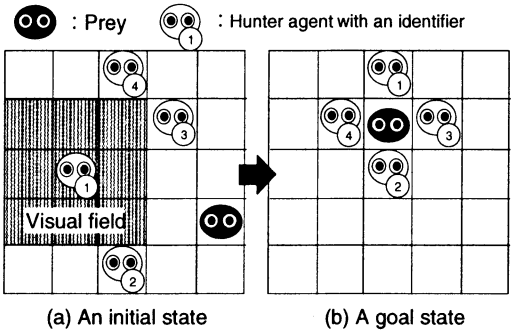


Figure 2: Pursuit Problem. A hunter is able to accurately recognize the relative position and the identifier of any other agents in its visual field. In Fig. (b), the prey is captured by the hunter agents.

In an $n \times n$ toroidal grid world, initially, a single *prey* and four *hunter* agents are placed at random positions respectively, as shown in Fig. 2(a). The final goal of the hunter agents is to capture the prey. The prey is judged to be captured when all of its four neighbor positions are occupied by the hunters, as shown in Fig. 2(b).

At every time step, each hunter independently chooses their own actions, not communicating with each other but using only sensory inputs. The prey chooses actions randomly. A hunter has a limited visual field and has a unique identifier as shown in Fig. 2. A hunter can recognize the identifier of any other agents within its visual field.

The sensory inputs to an agent are the relative positions to the prey and other hunter agents in the visual field. If the prey and/or some other hunter agent(s)

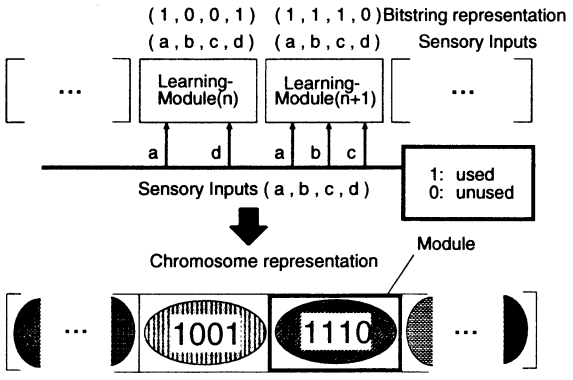


Figure 3: Representation

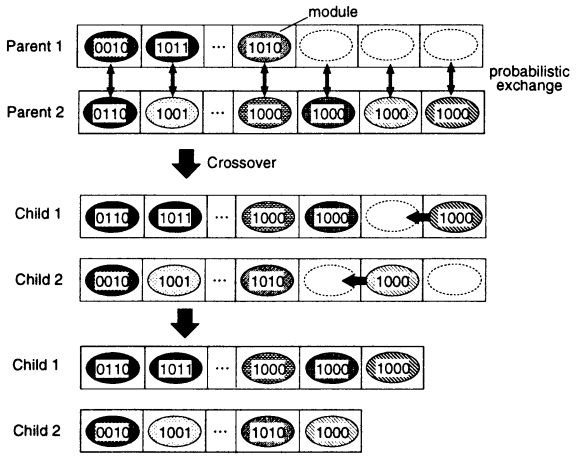


Figure 4: Crossover

do not exist in the visual field, the corresponding sensory inputs indicate that they do not exit in the visual field.

Each agent has a repertoire of five actions, that is, they can move in one principle direction (north, east, south or west), or remain at the current position. The prey can not occupy the same position that a hunter does, while hunters can occupy the same position.

3 Proposed Method

In this section, we propose an evolutionary method for finding the appropriate number and combination of modules for a given task. In this paper, we suppose that all the reinforcement learning agents in multi-agents world have homogeneous modular structures.

Representation : As shown in Fig. 3, a chromosome is an array of modules whose length is variable. A module is represented with the bitstring of n_p bits, where n_p is the maximum number of sensory inputs to an agent. The bitstring consists of 1 or 0, where 1/0 indicates that the corresponding sensory input is used/unused.

Crossover : As shown in Fig. 4, the crossover operator exchanges some modules between two parents. In the operator, the modules of Parent1 and Parent2 are probabilistically exchanged each other at each position on the chromosome. At the position where one parent has a module and the other does not, the module is moved to the tail of the chromosome of the other parent if the modules are determined to be exchanged at the position as shown in Fig. 4.

Mutation : We introduce two kinds of muta-

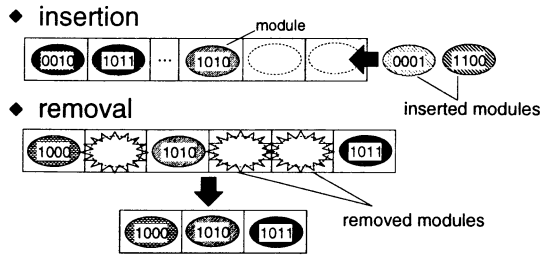


Figure 5: Mutation

tion operators, the insertion operator and the removal operator as shown in Fig. 5. The insertion operator adds some modules generated randomly to the tail of the chromosome. The removal operator deletes some modules from the chromosome randomly and moves the remaining modules to the left on the chromosome as far as possible. Either the insertion operator or the removal one is performed with a probability of 0.5.

Generation-alternation model : We adopt a variant of the Minimal Generation Gap (MGG)[Satoh 96] shown in Fig. 6 as the generation-alternation model. In the generation-alternation model used here, 1) two parents are randomly chosen from the population, 2) the crossover operator is applied to the parents to make two children with a probability of p_c and, here, if the crossover does not performed, the children are the copies of the parents, 3) the mutation operator is applied to the children with a probability of p_m, 4) the best two individuals are selected from the family consisting of the parents and their children, and 5) the parents in the population are replaced with the individuals selected at step 4.

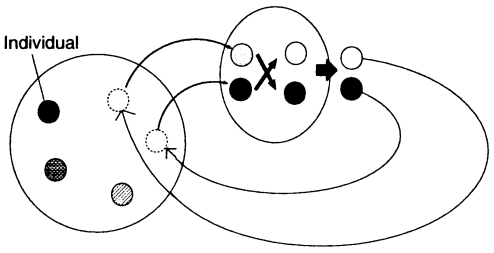


Figure 6: Minimal Generation Gap [Satoh 96]

4 Experiments

To demonstrate the effectiveness of the proposed method, we compared the performance of the modular structures found by the proposed method with that of the one designed by a human [Ono 96] on the Pursuit Problem [Benda 85].

4.1 Calculating Fitness

The fitness is calculated by the reciprocal of average time steps in which the hunter agents capture the prey. The average time steps are measured during the evaluation period after the learning period.

4.2 Parameter settings

We employ the Q-learning [Watkins 92] as a learning architecture as well as [Ono 96]. The differences between this experiment and [Ono 96] in the parameters of the Pursuit Problem and the Q-learning are that the initial Q-values are set to 0.1 in this experiment and that the hunters do not receive any negative rewards in this experiment.

A trial is aborted if the prey is not captured within 1,500 time steps. The environment is a 20 $\times$ 20 grid world, and each hunter agent has a 7 $\times$ 7 visual field.

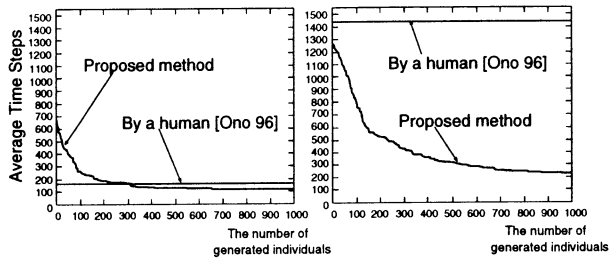
Because of the limitation on the memories of the computers, the number of modules of each agent is modified to a randomly generated number between one and five if it exceeds five and the number of sensory inputs used in each module is enforced to a randomly generated number between one and three if it exceeds three.

Upon capturing the prey, all the hunter agents receive rewards of 1.0. The learning rate, α, and the discount factor, γ, are set to 0.1 and 0.9, respectively. The initial Q-values of each module are set to 0.1.

The population size is 50, the crossover rate is 0.8 and the mutation rate is 0.05. The algorithm is terminated when the newly 1,000 individuals are generated in the population. An individual is evaluated for 100 trials after learning for 400 or 2,900 trials. We evaluate the performance of methods by the average of the result of five independent runs.

4.3 Results

Figure 7 shows the convergence curves of the proposed method when the learning period is 2,900(Fig. 7(a)) and 400(Fig. 7(b)). The horizontal axis shows the number of generated individuals and the vertical one does the average time steps to capture. The horizontal line with the label, "By a human [Ono 96]", shows the performance of the modular structure designed by a human in [Ono 96].



(a) The learning period; 2,900 trials (b) The learning period; 400 trials

Figure 7: Evolution process of modular structures

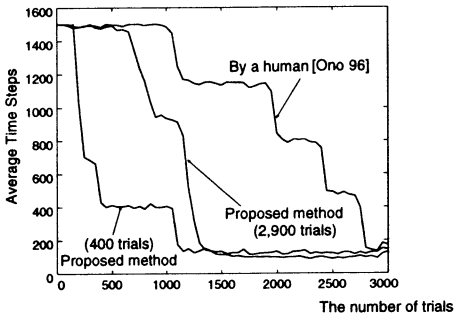


Figure 8: Learning process of modular structures

As shown in Fig. 7(a), when the learning period is long enough, the both modular structures designed by a human and found by the proposed method show good performance. Here, note that the performance of the proposed method is slightly better than that of [Ono 96] in the last search phases. On the other hand, when the learning period is short, the performance of [Ono 96] deteriorates considerably but that of the proposed method remains good, as shown in Fig. 7(b).

Figure 8 shows the learning curves of the modular structures designed by a human [Ono 96] and found by the proposed method with the learning period of 2,900 and 400 trials. The horizontal axis shows the number of trials and the vertical one does the average time steps to capture. The average time steps of [Ono 96] fall slowly while that of the modular structures found by the proposed method do very quickly.

We believe that the reason for the good performance of the modular structures found by the proposed method is that they have the module that observes only the prey as shown in Fig. 9. The module is not included in the modular structure designed by a human. We confirmed that the modular structure made by adding the module observing only the prey to the modular structure designed by a human showed al-

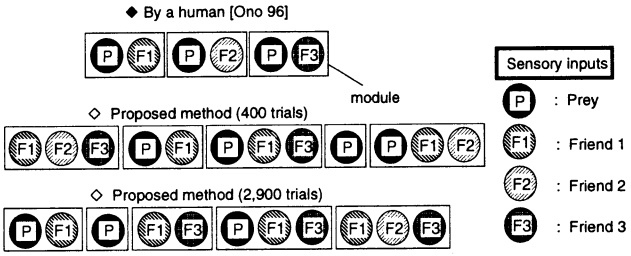


Figure 9: Examples of the modular structures designed by a human [Ono 96] and found by the proposed method

most the same performance as the modular structures found by the proposed method.

5 Conclusion

In this paper, we proposed an evolutionary method to automatically find appropriate modular structures for a given task in the MRLA. We show its effectiveness by applying it to the Pursuit Problem.

At the moment, the crossover operator exchanges two modules at each position on the chromosome. We are afraid that the exchange depending on the position on the chromosome prevents from generating various modular structures, which badly affects the search performance. We would like to propose a new crossover operator that does not depend on the position on the chromosome.

References

[Benda 85] Benda, M., Jagannathan, V. and Dodhiawalla, R.: On Optimal Cooperation of Knowledge Sources, Technical Report BCS-G2010-28, Boeing AI Center, 1985.

[Ono 96] Ono, N. and Fukumoto, K.: Multi-agent Reinforcement Learning: A Modular Approach, Proc. ICMAS-96, pp.252-258, 1996.

[Satoh 96] Satoh, H., Yamamura, M. and Kobayashi, S.: Minimal Generation Gap Model for GAs Considering Both Exploration and Exploitation, Proc. 4th Int. Conf. on Soft Computing, pp.494-497, 1996.

[Watkins 92] Watkins, C. J. C. H., and Dayan, P.: Q-learning, Machine Learning 8, pp.279-292, 1992.

[Whitehead 93] Whitehead, S., Karlsson, J. and Tenenberg, J.: Learning Multiple Goal Behavior via Task Decomposition and Dynamic Policy Merging, in J. H. Connell *et al.* Robot Learning, Kluwer Academic Press, 1993.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

An Evolutionary Approach to the Synthesis of Stable Grasp in Multi-Fingered Robot Hand

Y. Katada, M.Svinin, Y. Matsumura, K. Ohkura, and K. Ueda
Mechanical Engineering Department
Kobe University
Rokko-dai, Nada-ku, Kobe 657-8501, Japan

Abstract

This paper discusses grasping of objects by multi-fingered robot hand and proposes a procedure for the synthesis of stable grasp. In the procedure, the conditions for the Liapunov stablility and the contact stability are formulated as an optimization programming problem, and an evolutionary computation technique is applied to the problem. The effectiveness of the procedure proposed is illustrated on the computer simulation.

1 Introduction

The task of grasping and manipulating rigid objects using mult-fingered hand represents an enormous challenge, as witnessed by the considerable attention that the problem has attracted.

Analytical studies of grasping and manipulation by robot hands have been done by many researchers[1]. Kerr and Roth [2] proposed to determine the optimal internal force as that with the minimum norm under an approximated frictional constraint. Nakamura *et al.* solve the problem as a nonlinear programming problem using Lagrange multipliers [3]. Fukuda *et al.* proposed a method to plan the grasping points and the forces by using Evolutionary Computation [4].

In this paper, an evolutionary computation technique [6][7][8] is applied to the position/force planning for grasping an object by multi-fingered robot hand. The main feature of the problem is that not only the contact positions of the fingers but the contact forces, too, are used on the planning level. The contact points and forces should be specified in such a way so as to satisfy equilibrium conditions and unilateral frictional constraints. In addition, the contact point and the contact forces must satisfy the Liapunov stability conditions as well as the contact stability conditions. The stability constraints are introduced implicitly, through the local finger-tip gain coefficients and the maximal disturbances applied to the object.

This paper is organized as follows. In Section 2, Conditions on the Lyapunov stability and the contact stability of the object with unilateral frictional constraints are given. The computational procedure of Robust Evolutionary Programming is described in Section 3. REP is applied to the problem of position/force planning for grasping an object in Section 4. Finally, conclusions are presented in Section 5.

2 Formulation

Consider a rigid body subjected to multiple frictional contacts. Assume that constant forces $\boldsymbol{f}_1, \boldsymbol{f}_2, \ldots, \boldsymbol{f}_n$ are applied at the points defined by the radius-vectors $\boldsymbol{\rho}_1, \boldsymbol{\rho}_2, \ldots, \boldsymbol{\rho}_n$ drawn from the center of mass O. The body is at the equilibrium so the static equations read

$$\sum_{i=1}^{n} \boldsymbol{f}_i = -m\boldsymbol{g}, \quad \sum_{i=1}^{n} \boldsymbol{\rho}_i \times \boldsymbol{f}_i = \boldsymbol{0} \tag{1}$$

where m is the mass of the body and $\boldsymbol{g}$ is the gravity vector and n is the number of fingers. In the block form,

$$\begin{bmatrix} \boldsymbol{I} & \boldsymbol{I} & \cdots & \boldsymbol{I} \\ \Omega(\boldsymbol{\rho}_1) & \Omega(\boldsymbol{\rho}_2) & \cdots & \Omega(\boldsymbol{\rho}_n) \end{bmatrix} \boldsymbol{f} = \begin{bmatrix} -m\boldsymbol{g} \\ \boldsymbol{0} \end{bmatrix} \tag{2}$$

$$\Longleftrightarrow \boldsymbol{A} \cdot \boldsymbol{f} = \boldsymbol{b}, \tag{3}$$

here, Ω is the skew-symmetric operator such that $\Omega(\boldsymbol{a}) \cdot \boldsymbol{b} \equiv \boldsymbol{a} \times \boldsymbol{b}$.

The pseudo-inverse solution of (2) is given by

$$\boldsymbol{f} = \boldsymbol{A}^{+}\boldsymbol{b} + (\boldsymbol{I} - \boldsymbol{A}^{+}\boldsymbol{A})\boldsymbol{c}, \tag{4}$$

where $\boldsymbol{c}$ is an arbitrary vector.

The Coulomb friction constraints take following form:

$$| \boldsymbol{n}_i \times \boldsymbol{f}_i | \leq \mu(\boldsymbol{f}_i \cdot \boldsymbol{n}_i) \quad i = 1, \ldots, n, \tag{5}$$

where $\boldsymbol{n}_i$ is normal to the object's surface, pointing toward it, and μ stands for the static friction coefficient.

The stability conditions [5] are described as the following expressions:

$$\begin{aligned} L &\geq 0, \\ L^2 + S &\geq 0, \\ LS - V &\geq 0 \end{aligned} \tag{6}$$

where,

$$L = \sum_{i=1}^{n} \boldsymbol{\rho}_i \cdot \boldsymbol{f}_i, \tag{7}$$

$$S = \sum_{i=1}^{n} \sum_{j=i+1}^{n} (\boldsymbol{\rho}_i \times \boldsymbol{\rho}_j) \cdot (\boldsymbol{f}_i \times \boldsymbol{f}_j), \tag{8}$$

$$V = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \sum_{k=j+1}^{n} \{\boldsymbol{\rho}_i \cdot (\boldsymbol{\rho}_j \times \boldsymbol{\rho}_k)\}\{\boldsymbol{f}_i \cdot (\boldsymbol{f}_j \times \boldsymbol{f}_k)\}. \tag{9}$$

The conditions (6) are not always achievable under the unilateral grasping constraints. In such situations the feedback stabilization is necessary. Here we consider a simple control law of the following form:

$$\Delta \boldsymbol{f}_i = -k_p \Delta \boldsymbol{x}_i, \quad i = 1, \ldots, n \tag{10}$$

where $\Delta \boldsymbol{x}_i = \Delta \boldsymbol{x} + \boldsymbol{\theta} \times \boldsymbol{\rho}_i$ are changes in the desired contact positions. Mechanically, it corresponds to the introduction of virtual linear springs —with stiffness $k_p > 0$— at the contact points Fig 1.

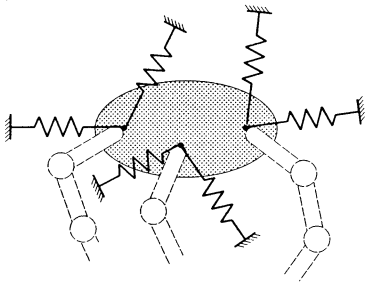


Fig. 1 Stiffness at finger-tip

The stability conditions under the feedback (10) are given as:

$$\begin{aligned} L' &\geq 0, \\ L'^2 + S' &\geq 0, \\ L'S' - V' &\geq 0 \end{aligned} \tag{11}$$

where,

$$L' = \sum_{i=1}^{n} \boldsymbol{\rho}_i \cdot \Delta \boldsymbol{f}_i, \tag{12}$$

$$S' = \sum_{i=1}^{n} \sum_{j=i+1}^{n} (\boldsymbol{\rho}_i \times \boldsymbol{\rho}_j) \cdot (\Delta \boldsymbol{f}_i \times \Delta \boldsymbol{f}_j), \tag{13}$$

$$V' = \sum_{i=1}^{n} \sum_{j=i+1}^{n} \sum_{k=j+1}^{n} \{\boldsymbol{\rho}_i \cdot (\boldsymbol{\rho}_j \times \boldsymbol{\rho}_k)\} \cdot \{\Delta \boldsymbol{f}_i \cdot (\Delta \boldsymbol{f}_j \times \Delta \boldsymbol{f}_k)\}. \tag{14}$$

Also the friction conditions (5) are written down in the following form

$$| \boldsymbol{n}_i \times (\boldsymbol{f}_i + \Delta \boldsymbol{f}_i) | \leq \mu \{(\boldsymbol{f}_i + \Delta \boldsymbol{f}_i) \cdot \boldsymbol{n}_i\} \tag{15}$$

Now, let us make the reduction to the optimization problem. Fitness function is set as following,

$$F = \sum_{i=1}^{n} \boldsymbol{f}_i^T \boldsymbol{f}_i + \sum_{i=1}^{n} k_{pi}^2$$

Finally, the optimization task can be formulated as follows: Minimize F, subject to (1)(5)(6)(11)(15), and

$$\begin{aligned} | \Delta \boldsymbol{f} | &\leq \Delta \boldsymbol{f}_{max}, \\ | \Delta \boldsymbol{C} | &\leq \Delta \boldsymbol{C}_{max}, \\ | \Delta \boldsymbol{x} | &\leq \Delta \boldsymbol{x}_{max}, \\ | \Delta \boldsymbol{\theta} | &\leq \Delta \boldsymbol{\theta}_{max}, \\ \Delta k_{p_i} &\leq \Delta k_{p_{max}}, \end{aligned} \tag{16}$$

where all the inequalities are understood in the component sense.

3 Evolutionary Programming

3.1 Robust Evolutionary Programming

The REP employed in this paper is described in the following way[6]. For $i = 1, 2, \ldots, \mu$, $j = 1, 2, \ldots, n$, and $k = 0, 1, \ldots, m$, each individual $\mathbf{X}_i$ is represented as:

$$\mathbf{X}_i = [\boldsymbol{x}_i, (\boldsymbol{\eta}_{i0}, \ldots, \boldsymbol{\eta}_{ik}, \ldots, \boldsymbol{\eta}_{im})] \tag{17}$$

$${}^t\boldsymbol{x}_i = (x_i(1), \ldots, x_i(j), \ldots, x_i(n)) \tag{18}$$

$${}^t\boldsymbol{\eta}_{ik} = (\eta_{ik}(1), \ldots, \eta_{ik}(j), \ldots, \eta_{ik}(n)) \tag{19}$$

$$x_i(j), \eta_{ik}(j) \in R$$

where $x_i(j)$ and $\eta_{ik}(j)$ denote the j-th component values of the vectors $\boldsymbol{x}_i$ and $\boldsymbol{\eta}_{ik}$, respectively. Notice that each $x_i(j)$ has $(m + 1)$ strategy parameters.

An offspring $\tilde{\mathbf{X}}_i$ is calculated in the following procedure. Firstly, three stochastic operations (O_{dup}, O_{del} and O_{inv}) are replaced the component values $\boldsymbol{\eta}_{ik}$ with $\boldsymbol{\eta}'_{ik}$. For $i = 1, 2, \ldots, \mu$, $j = 1, 2, \ldots, n$, $l = 1, 2, \ldots, m$, and $r \in \{1, 2, \ldots, m\}$, they are defined as follows:

$$\begin{aligned} O_{dup} \quad &: \quad \eta'_{i0}(j) = \eta_{i0}(j) \\ &\quad\quad \eta'_{il}(j) = \eta_{i(l-1)}(j) \qquad (20) \\ O_{del} \quad &: \quad \eta'_{i(l-1)}(j) = \eta_{il}(j) \\ &\quad\quad \eta'_{im}(j) = min(\eta_{max}, \sum_{k=0}^{m-1} \eta_{ik}(j)) \qquad (21) \\ O_{inv} \quad &: \quad \eta'_{i0}(j) = \eta_r(j) \\ &\quad\quad \eta'_r(j) = \eta_{i0}(j) \qquad (22) \end{aligned}$$

where η_{max} is a constant. Secondly, η'_i are modified as:

$$\tilde{\eta}_i(j) = \eta'_i(j) \exp(\tau' N(0,1) + \tau N_{ij}(0,1)) \qquad (23)$$

where $N(0,1)$ denotes a normally distributed one-dimensional random number with mean 0 and standard deviation 1. $N_{ij}(0,1)$ is a random number generated anew for every i and j. The factors τ and τ' are commonly set to $\left(\sqrt{2\sqrt{n}}\right)^{-1}$ and $\left(\sqrt{2n}\right)^{-1}$.

Then, the component values $\tilde{\boldsymbol{x}}_i$ are modified as:

$$\tilde{x}_i(j) \quad = \quad x_i(j) + \tilde{\eta}_{i0}(j)\delta_{ij} \qquad (24)$$

where we called $\eta_{i0}(j)$ active strategy parameters, and $\eta_{il}(j)$ inactive strategy parameters which are no effect of creating $\tilde{\boldsymbol{x}}_i$. The computational procedure of REP can be described in the same manner as FEP [8], as follows:

1. Generate the initial population of μ individuals, and set $g = 1$.
2. Evaluate the objective value for each individual $\mathbf{X}_i$, $\forall i \in \{1, 2, \ldots, \mu\}$ of the population based on the objective function $f(\boldsymbol{x}_i)$.
3. Each parent $\mathbf{X}_i, i = 1, 2, \ldots, \mu$, reproduces offspring $\tilde{\mathbf{X}}_i$.
4. Calculate the fitness of each offspring $\tilde{\mathbf{X}}_i$, $\forall i \in \{1, 2, \ldots, \lambda\}$, according to $f(\tilde{\boldsymbol{x}}_i)$.
5. Conduct pairwise comparison over the union of parents $\mathbf{X}_i$ and offspring $\tilde{\mathbf{X}}_i$, $\forall i \in \{1, 2, \ldots, \mu\}$. For each individual, $\mathbf{q}$ opponents are chosen randomly from all the parents and offspring with an equal probability. For each comparison, if the individual's fitness is no smaller than the opponent's one, it receives a "win".
6. Select the μ individuals out of parents $\mathbf{X}_i$ and offspring $\tilde{\mathbf{X}}_i$, $\forall i \in \{1, 2, \ldots, \mu\}$, having larger number of "win", to make them parents of the next generation.
7. End the procedure if the halting criterion is satisfied, otherwise, $g = g + 1$ and go to step 3.

3.2 Coding

In this computer simulation, the given data are the shape and mass of the object and the friction coefficient between the object surface and the finger. The task is grasping of the circular object by 4 fingers in 2D and the cylindrical object by 3 fingers in 3D.

As shown in Fig. 2,3, an individual is defined as a row of a real number, which means the height and angle on the object, an arbitrary vector in the force derived from (4), local finger-tip gain coefficients, and the maximal disturbances on all fingers.

The individual size are 18 in 2D and 24 in 3D.

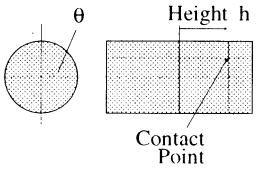


Fig. 2 Contact Point on the Cylinder

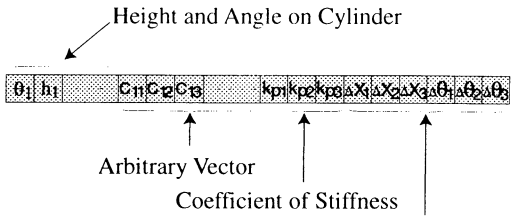


Fig. 3 Coding of Individual

4 Simulation Result

We conducted simulation for defferent values of μ and m. The results of four simulations can be summarized as follows;

[Experiment 1](2D case) $\mu = 0.25$. The results are kp_1=11.218,kp_2=83.120,kp_3=28.446,kp_4=15.299. $\Delta\boldsymbol{x}$={ -23.485,-24.169 }. $\Delta\theta$= 0.068(rad).

[Experiment 2](2D case) $\mu = 0.9$. The results are kp_1=43.097,kp_2=22.890,kp_3=0.175,kp_4=52.883. $\Delta\boldsymbol{x}$={ 26.856,-10.213 }. $\Delta\theta$= -0.018(rad).

[Experiment 3](3D case) $\mu = 0.9, m = 5$. The results are kp_1=8.056,kp_2=19.436,kp_3=47.029. $\Delta \boldsymbol{x}$={ 8.056,-7.100,3.179 }. $\Delta \boldsymbol{\theta}$={ 0.379,-0.526,-0.101 }(rad).

[Experiment 4](3D case) $\mu = 10.9, m = 5$. The results are kp_1=18.058,kp_2=36.678,kp_3=80.926. $\Delta \boldsymbol{x}$={ 16.323,12.363,13.564 }. $\Delta \boldsymbol{\theta}$={ -0.105,1.213,1.558 }(rad).

Fig. 4-7 show the configulation of fingers from the result of the simulation. The arrows mean the position and the force vector of fingers (the length means relative scale among forces).

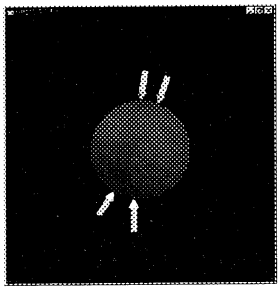


Fig. 4 Positions and forces ($\mu = 0.25$)

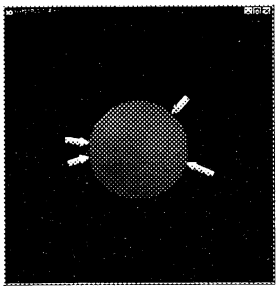
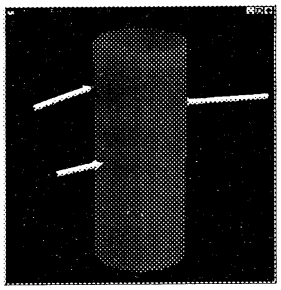


Fig. 5 Positions and forces ($\mu = 0.9$)



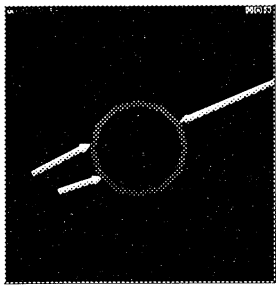
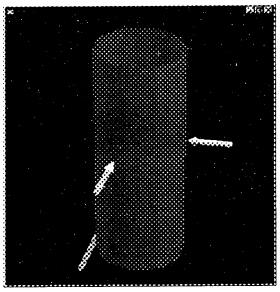


Fig. 6 Positions and forces ($m = 5, \mu = 0.9$)



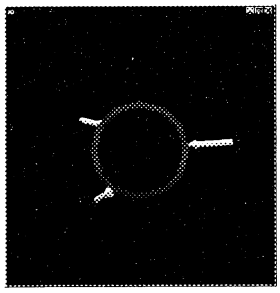


Fig. 7 Positions and forces ($m = 5, \mu = 10.9$)

As seen in Fig. 4-7, a pair of fingers' positions tend to gather. It can be thought as the effect of the fitness function.

5 Conclusions

The optimization problem on grasping force and position by using Robust Evolutionary Programming has been considered in this paper. Computer simulations were conducted on the circular object by 4 fingers in 2D and the cylindrical object by 3 fingers in 3D. In evolutionary process, force and position of the fingers are decided, and the stiffness of finger-tip and the range of disturbance where the system can stand being stable are calculated. The solution which are gotten from the simulation satisfies all constraints.

Acknowledgements

This study has been conducted in "Methodology of Emergent Synthesis" Project (JSPS-RFTF96P00702) supported by the Japan Society for the Promotion of Science.

References

[1] K. B. Shimoga, "Robot Grasp Synthesis Algorithms: A Survey," *The International Journal of Robotics Research*, Vol. 15, No.3, pp. 230-266, 1996.

[2] J. Kerr, B. Roth, "Analysis of multifingered hands," *The International Journal of Robotics Research*, Vol. 4, No.4, pp. 3-17, 1986.

[3] Y. Nakamura, K. Nagai, T. Yoshikawa, "Dynamics and stability in coordination of multiple robotic mechanisms," *The International Journal of Robotics Research*, Vol. 8, No.2, pp. 44-61, 1989.

[4] T. Fukuda, K. Mase, Y. Hasegawa, "Robot Hand Manipulation by Evolutionary Programming," Proceedings of the 1999 IEEE International Conference on Robotics & Automation, pp. 2458-2463, 1999.

[5] M. M. Svinin, K. Ueda, M. Kaneko, "On the Stability of an Object in Multi-Finger Grasping," Proceedings of the 1999 IEEE International Conference on Robotics & Automation, pp. 257-262, 1999.

[6] Y. Matsumura, K. Ohkura and K. Ueda , "Evolutionary Programming with Non-coding Segments for Real-valued Function Optimization", *Proc. of 1999 IEEE International Conference on Systems, Man and Cybernetics (SMC'99)*, vol.4, pp.242-247, 1999.

[7] D. B. Fogel , *Evolutionary Computation Toward a New Philosophy of Machine Intelligence*, IEEE Press, 1995.

[8] X. Yao, Y. Liu and G. Lin , " Evolutionary Programming Made Faster", *IEEE Transactions on Evolutionary Computation*, Vol. 3, No. 2, pp.82-102, 1999.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Evolution of multi-agent system strategies using genetic programming

Masanori Sugisaka* and Xiaoshu Wang**
Dept. of Electrical and Electronic Engineering
Oita University, 700 Dannahara,Oita 870-1192, Japan*
{wangxs*,msugi**}@cc.oita-u.ac.jp

Abstract

Learning and evolving in artificial agent is an extremely difficult problem, but on the other hand, a challenging task. At present the studies mainly centered on single agent learning problem. In our case, we use simulated soccer to investigate multi-agent cooperative learning. Consider the fundamental differences in learning mechanism, existing reinforcement learning algorithms can be roughly classified into two classes-that based on evaluation functions and that of searching through policy space in direct. Genetic Programming developed from Genetic Algorithms is one of the most well known approaches that belong to the latter. In this paper, we give detailed algorithm as well as data construction description that are necessary for learning single agent strategies at first. In the following sections, we extend developed methods into multiple robot domains moreover. We investigate and contrast three different solutions-single agent learning, simple team learning and sub-group learning and conclude the paper with some actual experiments and result analyses.

Keywords: Genetic Programming, Multi-agent system, Machine Learning

1 Introduction

Learning is the problem faced by an agent of how to obtain optimal problem-resolving policy through trial-and-error interactions with a dynamic environment. Agent's behaviors can be learned from two kinds of information. First in the case of agent associations can being provided., supervised learning takes place. Secondly, if existing agent's behaviors can be evaluated. Here, reinforcement learning provides an appropriate framework. Consider the basic differences in learning mechanism, we can roughly classify existing reinforcement learning algorithms into two classes. The first includes those that base on adaptive Evaluation Functions (EFs) such as Q-learning presented by Watkins in 1989 and all kinds of variants of it. Methods from the second class do not require Efs at all. They search through the policy space directly so as to obtain optimal policies. One of the most widely known examples of EFs-free approches is Genetic Programming(GP) proposed by Koza1 in 1992. For explanation in more detail, a survey of RL written by Leslie[2] et al is a good reference, I think.

Originally all of them were developed for solving single agent learning problems. But recently multi-agent learning drew more and more attentions in several research areas. In our case, we use simulated soccer to investigate multi-agent cooperative learning.

2 Machine Learning

At first let's investigate machine learning in general. Problems with delayed reinforcement are well modeled as *Markov Decision Processes* (MDPs). An MDP consist of

1) a set of states S,
2) a set of actions A,
3) a reward(or cost) function $R: S \times A \rightarrow IR$,
4) a state transition function $T: S \times A \rightarrow \Pi(S)$.

where $\Pi$is a discrete probability distribution over set S.

Consider a MDP model shown above, the goal of learning becomes to search optimal policy to maximize expected reward (or, minimize total expected discounted cost). Let's Denote policy as π, then the optimal polity based on MDPs model can be described as

$$V^*(s) = \max_a(R(s,a) + \gamma \sum_{s' \in S} T(s,a,s')V^*(s'))), \forall a \in A$$

$$\pi^*(s) = \arg V^*(s),$$

where $R(s,a)$ is the instantaneous reward, s' resulting state, and $T(s,a,s')$ the probability of taking action a in state s. V is evaluation function of a state. In other words, it is a *value* that agent can gain if it starts in state s and executes a policy π.

If the state transition probability function $T(s,a,s')$ and the reinforcement function $R(s, a)$ are known, that is, the model is known. The problem can be well solved by several effective algorithms, such as *value iteration iterative algorithm*[2]. But in general we can not know these informations in advance. So the agent must interact with its environment directly to obtain information.

A conceptually simple solution is Q-learning presented by Watkins in 1989. He defines

$$V^*(s) = \max_a Q^*(s,a), \forall a \in A,$$

and

$$Q^*(s,a) = R(s,a) + \gamma \sum_{s' \in S} T(s,a,s') \max_{a'} Q^*(s',a')),$$

Where $Q^*(s,a)$ is the expected reward (or discounted gain) of taking action a in state s, which is defined as

$$Q(s,a) := Q(s,a) + \alpha(r + \gamma \max_{a'} Q(s',a') - Q(s,a)).$$

$Q(s,a)$ can be learned using artificial neural network and so on.

In this several years, researchers take more and more interest in solutions of searching through optimal policies in policy place directly. Among them Genetic Programming(GP) proposed by Koza is most well known. In fact GP is a branch of Genetic Algorithms where the solution strings are replaced with variable length programs. Sugisaka[3] and Wang[4] show basic thoughts of how to apply Genetic Algorithms to robot control in some papers before. The goal of GP is to find a program that can be run so as to solve practical problem. Such problem-solving program in fact stands for policy π in MDPs model shown above. Then the problem of learning becomes that of how to evolve programs so as to satisfactory results (optimal policies) can be found out finally. From now in this paper PROG denotes a genetic program. The genetic program is evaluated for fitness by executing PROG. Usually PROGs are decoded in format of n-ary tree with n being the maximal number of sub-trees that a non-leaf node can contain. A non-leaf node encodes a function f_i. A leaf node stands for a terminal t_i. So in order to initialize PROG population, a function set $F=\{f_1,f_2,\ldots,f_k\}$ with k functions and a terminal set $T=\{t_1,t_2,\ldots,t_l\}$ with l terminals must be given definitely in advance. GP follows same steps as GA but the steps act a little differently in order to make them reasonable to the new tree-type structure. The flow of GP operation is shown in figure 1.

```
Initialize the Population of PROGs arbitrarily
Loop until termination criterion satisfied or timeover
Evaluate PROGs in population
Loop until next generation population created
Select genetic operation probabilistically
Perform Reproduction
Perform Crossover
Perform Mutation
End loop
End loop
```

Fig.1. Genetic programming flowchart

Given the MDPs model, the set of action A can be used as terminal set T directly. But the set of function must be designed carefully so as to it can exploit environment well. In this paper, we will use GP to solve robot soccer learning problem.

3 Learning soccer strategies by GP

In this years robot contest becomes a popular problem area for studies in artificial intelligent and machine learning, such as those conducted by John WS[5] and Ho F[6] et al. In our work we use simulated soccer to study intelligent multi-agent system. The reasons for us choosing such a system as experiment bench are because (1) it is a standard multiple agent system suitable to be studied, (2) it is of flexibility. In other words, we can expand or contract the scale of system easily to meet our needs. (3) it is ideal to tell whether an approach better than another or not since it is a kind of antagonistic game played by two teams.

Initially our works in soccer robot are limited to constructing fixed strategies. But with the number of robot players increasing, huge complexity of state space makes it impossible for us to consider all conditions the agent may encounter in actual playing. That is to say, pre-planned strategies are not adaptive to practical environment change. Efforts for solving this problem bring about ideas of learning robot soccer game strategies. Especially we put our emphasis on how to co-learn multiple agents simultaneously.

3.1 Robot soccer

Robot soccer game, denoted *G{Field(Size, CoordinateSystem), ball, Teamhome, TeamOpp, Rule}*, is played by two robot teams, *Teamhome* and *TeamOpp*. Each team is composed of n robots, that is, $Team=\{P_1,\ldots,P_n\}$ $n>1$. The game rule, *Rule*, comes from human soccer sports. The ball is described as *ball{b(x,y), direction, Velocity}*. *b(x,y)* is center coordinates of the ball. The ball moves in *velocity* along orientation of *direction*. A player contains three same variables of P_i *(x,y)* and *direction, velocity*, and more one variable, *Size(l,w)*, is included also. For GlobeVisionSystem-based micro-robot system, a Boolean variable for ball and each player is added in since in some time the ball or players can not be identified during playing. At any sampling time point, a player can obtain information *i(p,t)* about present environment. *i(p,t)* includes 1) $ball_t$, 2) *Teamhome* and *TeamOpp* 3) home players' behaviors in last sampling period. Of course the information of *Field* is also known since it is treated as global constant. With the number of players increasing, the environment information *i(p,t)* will become larger and larger. Processing of *i(p,t)* will become very time-consuming. In this case, we can just consider the information belonging to the round with the radius of r and center

coordinates of *p(x,y)*, that is, it assumes that the behavior of a player mainly is affected by its circumference in nearby.

There are several kinds of methods to organize an system to attend contest. They are (1) simulation game in computer, (2), (3) simple distributed robot system, (4) humanoid robot system and so on. In our case we conduct simulations in computer at first. Then we test obtained optimal strategies in GlobeVisionSystem-based micro-robot system.

3.2 Learning Single robot strategy

We start with single robot learning, that is, there are neither partners nor opponents. A simulated robot in the play field can execute any action from action set A. An example of set A is shown in table 1.

Table 1. An example of Action set.

A_1	*go-forward*	Move player *p* one step forward along its current direction, *(one step=Field.size.length/100)*
A_2	*go-backward*	Move player *p* one step opposite to its current direction
A_3	*drip-ball*:	If player have the ball, it will move with ball in one step. Otherwise it will execute action *go-forward.*
A_4	*kick-ball*	If player have the ball, it will move execute *go-forward* and at same time the ball obtain energy *E*. the speed of ball will be adjusted. Otherwise player just execute action *go-forward.*
A_5	*Trun-to-ball*	Change player's direction so that it directs to ball or opponent goal or position(x,y).
A_6	*Turn-to-goal*	
A_7	*Turn-to-there*	
A_8	*noise-shoot*	If player does not have the ball then do nothing. Otherwise, execute *turn(noise)* at first and then kick-ball. (This action is added in order to mimic the practical soccer game).

Except those actions listed in Table 1, we also tried some other actions, such as avoid-obstacle and shoot-to-goal. More actions was included, of course larger the PROG would became. But from actual experiment results we can not find that better performances appear if we just add more actions into action set.

In simulations, we designed all members *f* but a special function Proc2 in the function set *F* as two-branch conditional statement of IF-Then. In a condition, the left node is executed if the condition is TRUE, otherwise the right node is executed. Proc2 is not a condition. Both children nodes of it will be executed. Here we don't want to list all members included in the function set *F* since we continue trying new functions during practical experiment. We give some examples in Table 2.

Table 2 An example of function set.

F_1	**Proc2** (Note:a special function, both of its children nodes are executed.)
F_2	**IF** ball in hand **THEN**
F_3	**IF** obstacle ahead **THEN**
F_4	**IF** ball is near to opponent goal **THEN**
F_5	**IF** ball is near to home goal **THEN**
F_6	**IF** ball is between player with opponent goal **THEN**
F_7	**IF** ball is between player with home goal **THEN**

Based on the GP flowchart shown in Figure 1. GP algorithm works as follows.

1. To Initialize the Population

In the case of GP, to initialize a population means to generate n *2*-ary trees. Normally a *2*-ary tree can be created by recursive algorithm. That is, at first a member from set *AUF* is picked out randomly and inserted in current position. If selected member belongs to set *F* then left sub-tree and right sub-tree of it are created separately. In our case, the probability of selecting each member is same. So this method will work only if there are more than twice as many Actions as Functions. Otherwise the tree maybe become infinitely big. By the way, we keep the depth of the tree below 10. An example of generated tree is

shown in Fig. 2.

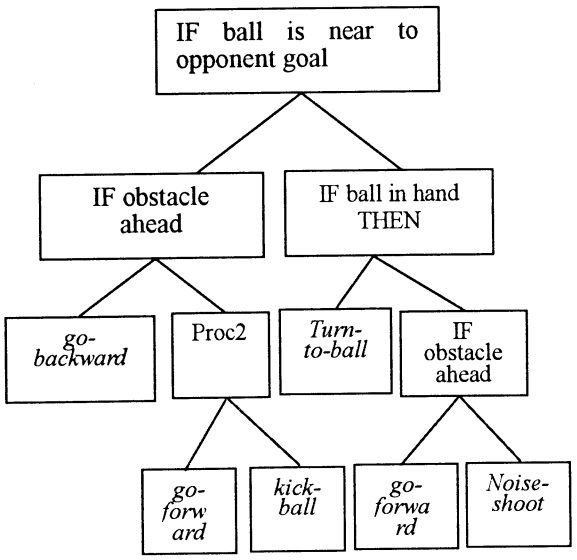


Fig. 2. An example of PROG tree.

2. To evaluate PROGs in population.
In our case, we execute all PROGs in current population in order to evaluate them. One entire soccer game lasts 1 minute. When game is over, every PROG is scored with a scalar, non-negative fitness value, *Fitness(PROGi)=number of goals scored by PROGs.*

3. To perform GP operations(Reproduction, Crossover, Mutation)
Reproduction operation runs the same for a GP as a GAs. The best *m* PROGs in current population are reproduced in order to replace the worst *m* PROGs. But the crossover and mutation operation must be re-designed to be suitable for tree-type genome. For crossover operation, at first a crossover node in each parent PROG tree must be selected. Unlike GAs, the crossover points in both parents do not have to be the same. For mutation operation, the leaf nodes and non-leaf nodes are not interchangeable. A non-leaf node of action member obviously leads grammar error.

GP is a general problem-solving method. Basic operations of it are independent to practical problems. In order to re-use the programming codes, GP can be broken down into problem-dependent and problem-independent parts with a carefully designed interface between these two parts. Usually the interface is a kind of structured input/output files. Some general GP modules, for example the GP kernel developed by Helmut H[7] at Vienna university, can be downloaded in free for research. In our laboratory, we developed our own GP VBX using C++.

3.3 Learning multiple robots strategies

Robot soccer contest is a kind of multiple robot tournament. In our case we try (3 by 3) and (5 by 5) games. Obviously we can extend single-agent GP learning into multi-agent area in 2 approaches. The simplest one is to learn all home player strategies separately where partners in the play field are treated as part of environment. We name it simple team learning. In this case except those basic information such as position and speed, any player can not know what his partner had done in last step and what he want to do in next step. Since now 6/10 robots are involved in the game, collisions between robots become so frequent that robots always gather around the ball and in same time the game can not continue at all. So we have to add collision-examination functions into *F* and collision-avoidance actions into A.

Another approach named sub-group learning treats all home players as an indivisible group or several divided sub-groups. For learning team/sub-group strategies, some changes must be made on existing single agent learning algorithms. Now the actions in set A ought to could control not only single robot but also several robots simultaneously, that is, a communication channel ought to be provided among agents who are in same sub-group. For example new action pass-ball will let one robot kick ball as well as move another robot to suitable position so that it can get the ball.Because opponents are involved in the game, the fitness function was changed to

Fitness(PROGi)=100+number of goals scored by PROGs-number of goals scored by opponent .

The offset 100 ensure the fitness always positive.

3.4 Experiments and results

We conduct three different types of simulations-single robot using single-agent learning, 3 by 3 using simple team learning and 5 by 5 game using sub-group learning separately. The parameters for GP in the simulations are listed in Table 3. The opponent team is driven by our old version of pre-programmed strategy.

Table 3. Parameters.

Team	1	3 by 3	5 by 5
Population size	100	50	50
Generation	2000	2000	2000
Maximum depth	10	10	10
Size of set *A*	10	14	16
Size of set *F*	4	6	7
crossover rate	0.2	0.2	0.2
mutation rate	0.05	0.05	0.05

Some experiment results are given in Figure 3.From results, we can find single agent learning works quite well. In the 1000th generation, the satisfactory policy can be generated. But in 3*3 game, home team can not win the game at all. So it seems that the simple team learning can not automatically generate

cooperative behaviors among agents. In 5 by 5 game, because the cooperation behaviors are considered in advance when we design action set and function set. The performances of learning become better, but it is not best result, I think.

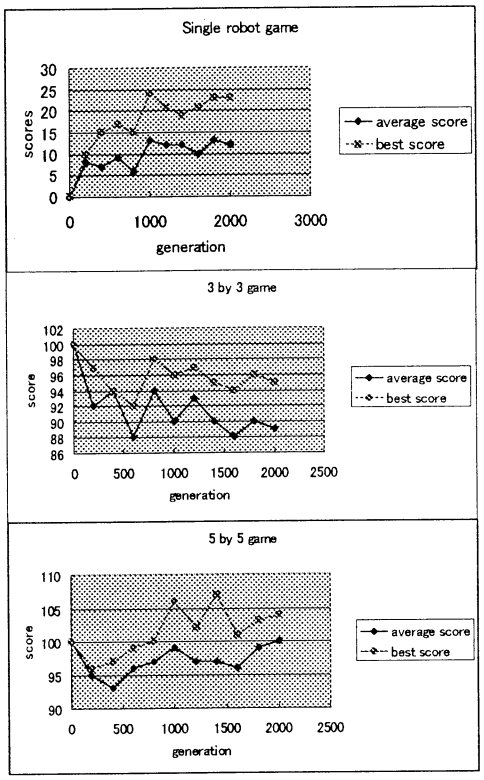


Fig. 3 Experiment results, from above
a)single robot game b)3x3 game c)5x5 game

4. Conclusion

We use simulated soccer contest to investigate multi-agent learning. In this paper, we compared single agent learning, simple team learning and sub-group learning based on GP. From results we know that single agent learning by GP works well. But in simple team learning case, expected cooperative behaviors among home players did not appear. In 5 by 5 game, we divide the home team into 3 sub-group of {2(attacker), 2(defender), 1(goalkeeper)}. Have action set A been designed carefully, the sub-group learning works better than simple team learning. But some problems still exist. For example if the number of player increases, the designing of action set becomes very awkward. In our next works we will 1) improve sub-group learning algorithms and 2) try EF-based RL algorithms and compare it with EF-free approaches

References

[1] J.R.Koza: Genetic programming-on the programming of computers by the natural selection, *Cambridge, MA, MIT Press* (1992)

[2] P.K.Leslie & L.L.Michael: A survey of reinforcement learning, *Journal of Artificial Intelligence Research 4*, 237/285(1996)

[3] M.Sugisaka & X.S.Wang XS & J.J.Lee: The genetic algorithms to evolve multiple agent cooperative system. *Proceedings of the International Symposium on Artificial Life and Robotics*, Beppu, Oita, Japan, pp.178/182(1998)

[4] M.Sugisaka & X.S.Wang: The autonomous evolution of cooperation activity of micro-robots by genetic algorithms. *Proceedings of the 29th ISCIE International Symposium on Stochastic system and its Applications*, Tokyo, Japan, pp.29/32(1997)

[5] W.S.John: Colearning in different games. *Machine Learning 33*:201/233(1998)

[6] F.Ho & M.Kamel: Learning coordination strategies for cooperative multi-agent systems. *Machine Learning 33*:155/177 (1998)

[7] H.Helmut: A C++ class library for genetic programming: *the Vienna university of economics genetic programming kernel-Operating Instruction*, 1:1/69(1996)

[8] C.Guy & B.George & D.N.Brigitte: Structure properties and classification of kinematics and dynamic models of wheeled mobile robot. *IEEE trans. on robotics and automation12(1)*, 47/61 (1996)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Emergence of Individuality and Sociality by Reinforcement Learning in Multi-Agent Systems

Katsunari Shibata, Masahide Ueda, & Koji Ito
Dept. of Computational Intelligence & Systems Science, Tokyo Institute of Technology
4259 Nagatsuta, Midori-ku, Yokohama 226-8502, Japan. shibata@ito.dis.titech.ac.jp

Abstract

In this paper, a new viewpoint, "individuality" and "sociality", is introduced for analyzing the multi-agent system's behavior. A hypothesis is set up that the both behavioral characters emerge to generate different actions from the other agent to increase each agent's benefit. "Individuality" is defined as the difference of actions between agents based on the difference of its internal information processing. While "sociality (rule)" is defined as the difference of actions based on the difference of its sensory inputs.

Next, a model is proposed in which "individuality" and "sociality" are obtained by reinforcement learning. It is also mentioned that there exist some factors like asymmetry of the environment, which influence the differentiation into one of the two characters. Finally through some simulations of conflict avoidance problems among passengers getting on and off a train, it is examined that the differentiation is adaptive to some of the above factors appropriately, and the rule that the passengers getting off have a priority to go.

1 Introduction

When we see our human society, we make and obey many rules, while each of us have a variety of individualities. Too many rules reduce the effectiveness of the society like a signal at the cross with less traffic. On the other hand, excessive individualities break the order of the society. Both are the opposed characters with each other, and necessary but must be moderate.

In some works in multi-agent(robot) systems and game theory, some individualities are given and the ratio of the agents having each individuality changes adaptively like the Hawk-Dove model[1] and so on [2]. Ota et al. proposed a distributed strategy-making using the instant reinforcement signals[3]. A kind of rule that was changed adaptively according to the given environment, emerged among the agents based on each agent's learning by only its own reward. However, they did not mention "individuality" and "sociality" at all, and the given task in their work is too simple in the following meanings. The reinforcement signal is given for each decision, and the number of the states and the number of actions for each agent are only two. Furthermore no serious conflict is assumed. The serious conflict means that in order to avoid the conflict, an agent has to take a strategy which is bad when the opponent agent does not exist.

In this paper, the existence of individuality and society is not dealt with as a premise, but the emergence of the both characters itself is also a subject. At first, "individuality" and "sociality" are introduced, and the factors of the differentiation into one of the two are considered. Then some simulations are done with more realistic environment.

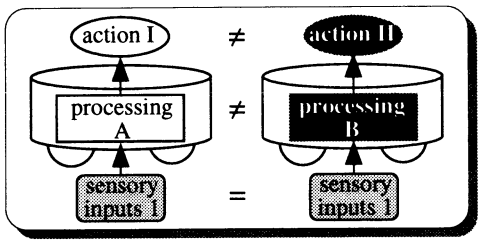
2 Individuality and Sociality

2.1 Hypothesis and Definition

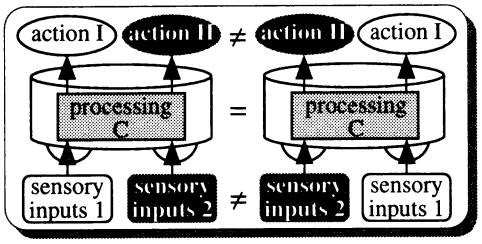
A hypothesis set up here is that both "individuality" and "sociality" are not inherited, but are emerged through each agent's learning to increase its benefit. The latter is the same standpoint as Adam Smith's[4].

Both behavioral characters are defined between two agents at first. Fig. 1 shows the definitions. "Individuality" is defined that each agent generates a different action from the other agent's even if the sensory inputs are the same. While, "sociality" is defined that two agents generate the same action when the sensory inputs are the same, but they generate different actions when the sensory inputs are different. If each agent's action is observed to be exchanged with each other when their states were exchanged, it is called that there exists "sociality" between them. If the action is the same, it is called that there is "individuality" between them.

In multi-agent case, it is not easy to define each agent's behavioral character. Here, for the present, it is judged by observing all the relations to the other agents. If "individuality" is the relation with more agents than "sociality", it is defined that "individuality" is stronger in the agent. In the contrary case, the agent is defined to have strong "sociality".



(a)individuality



(b) sociality

Figure 1: Definition of "identification" and "sociality" between two agents.

In the task (environment) in which two agents cannot achieve their purposes or can get less reward when they do the same action, they are expected to generate the different actions and avoid such situations by applying reinforcement learning to each agent. A model is proposed that the both characters emerge through such experiences and learnings

2.2 Differentiation and Its Factors

As in the previous section, there are two ways for an agent to generate a different action from the other agent. The reinforcement learning can decide which behavioral character is appropriate in the relation to the other agents. The factors which affect the differentiation can be picked up as follows. (1) Asymmetry of the environment, (2) Identification of the other agents, (3) Physical and mental characters of each agent, and (4) Communication with the other agents. If the environment is so asymmetric that it influences the agent's performance, some rules emerge, but the environment is symmetric, the agent can utilize only "individuality" to generate the different action. If the other agents can be identified, the agent can change its strategy according to the other, and "individuality" is apt to emerge. If an agent's physical or mental character is different from the others, it is easy for the agent to generate the different actions based on "individuality". The communication with the other agent helps to identify the other, and that results in the emergence of "individuality" [5]. In this paper, it is examined if some of those factors influence the differentiation appropriately.

3 Simulation

In this paper, the problem in which passengers are getting on and off a train as shown in Fig. 2 is taken as examples. In this problem, when both the passengers getting on and getting off are going to go straight, conflicts may happen. It is well-known that the rule is effective that the passengers getting off have priority to go.

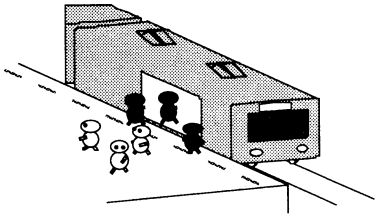


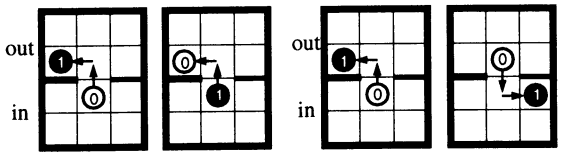
Figure 2: Conflict avoidance problem of passengers getting on and off a train.

3.1 Experimental Setup

Whole the space is represented as a grid world as shown in Fig. 3, and the boxes are divided into two regions, "in the train" and "out of the train". Some agents are initially located randomly in the stated places in the both regions. Only one agent (passenger) can exist in a box, and the agent can move only to the next box. If the number of whole the agents is larger than that in one simulation run, the agents for each run are chosen randomly. The order that the agents action depends on the initial locations.

The number of sensory inputs for each agent is 8. The first input is the role of the agent which represents whether the agent is "getting on" or "getting off". That depends on the initial location. The second input represents whether the doorway exists "in front or at the back", "right hand side", or "left hand side" of the agent. The states of the next boxes are indicated by the following four inputs. The state can be one of the four states, "empty", "occupied by the same group agent", "occupied by the opponent", and "wall exists between the boxes". Finally the other two inputs represent the state of the front box from the view of the right-next and left-next boxes respectively.

Each agent can select one of 5 actions, "move to each of the four next boxes", and "does not move". If there exists some agent already in the destination box, it does not move. If each agent arrives at its goal, it stops to learn and move randomly while restricted to move only in the goal area. The process until all the agents arrive at their goals, is defined as one trial, and one cycle means that all the agents move once.



(a) sociality (rule) (b) individuality

Figure 3: Example of "sociality" and "individuality" between two agents in the passengers problem.

In this simulation, one of the agents facing each other has to give its way to the other to avoid the conflict. It is observed which agent gives its way, and after the place of one agent is exchanged to the other's, it is observed again. Then if the different agents gives their ways as shown in Fig. 3 (a), there exists a rule and "sociality" exists between them. If the same agent gives its way as shown in Fig. 3 (b), the relation is judged as "individuality". The agent who gives its way has "individuality" of seeming "altruistic", while the other has "individuality" of "selfish".

3.2 Learning

Here, one-step Q-learning is employed as a reinforcement learning algorithm. The agent can get a reward 1.0 when it arrives at its goal. The goal is the upper end for the agent going upward and the bottom for the agent going downward. It gets no reward by the other agent's goal. That assumption is based on the selfish agent hypothesis as in the subsection 2.1. As each agent's strategy of action selection, Boltzmann Selection is employed, and the temperature is reduced exponentially from 1.0 to 0.01 during learning. The learning rate is 0.01 and the discount factor γ is 0.92.

3.3 Results

At first, the adaptive differentiation according to the specification of the other agent and the asymmetry of the environment is examined in the two-agent case. In this case, two agents are chosen randomly among some agents, and the choice and trial are repeated 40000 times. The grid world consists of 4x5 boxes, that is the same as Fig. 4 (a). In the case that the agents identify the other agents, the state of the next box can be not only opponent or wall, but can be agent1..6 except for the agent itself. This means that if the opponent is different from the previous trial, the state is perfectly different, and the previous learning does not affect the present learning at all. When the number of whole the agents is 2, the identification does not have any meaning because the number of the opponent is only one. In the symmetrical environment,

Table 1: Acquired behavioral character after learning in the case of two-agent. 5 simulation runs are done for each case. (ind: all the agents acquire "individuality", soc: all the agents acquire "sociality", mix: some agents acquire "individuality" and the others acquire "sociality", and fail: the task fails for some combinations of the agents)

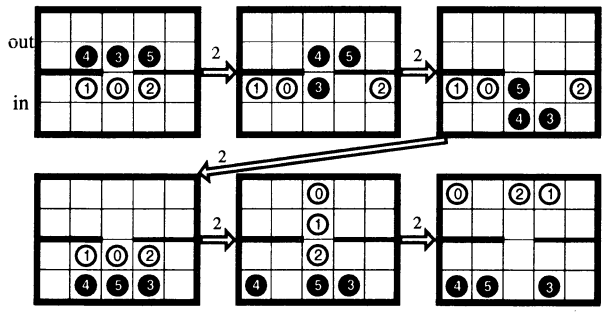
	number of whole agents	identify	non-identify
symmetrical inputs	2	ind 5	
	3	ind 5	mix 3 fail 2
	4	ind 5	fail 5
asymmetrical inputs	2	ind 3 soc 2	
	3	ind 1 mix 4	soc 5
	4	mix 5	soc 5

the agents are assumed that they cannot know whether they are getting on the train or getting off the train.

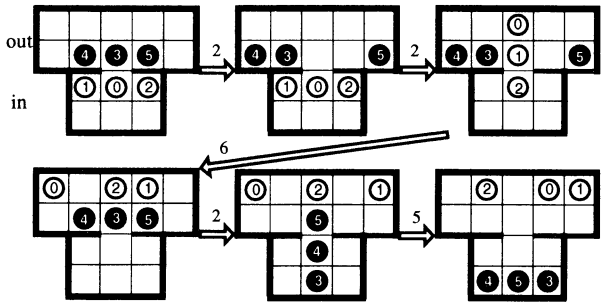
Table 1 shows the number of the acquired characters after 5 simulation runs for each case. It can be seen that in the case of non-identification and asymmetrical environment, "sociality" is apt to emerge, while "individuality" is apt to emerge in the case of identification and symmetrical environment. In the case of non-identification and symmetrical environment, some agents fail to arrive at its goal. The reason is that if both selfish and altruistic agents exist, the third agent has no effective solutions. Further, if they made a rule that the agent who moves first gives its way, the system becomes unstable. It is always profitable for an agent not to obey the rule and to stay at the same location, because the opponent agent always gives its way at the next time step.

Next, it is examined in six-agent simulations whether the rule emerges that the passengers getting off the train have a priority to go. The experimental setting is almost the same as the previous simulation, but the number of trials is 100000. At first, the case of a symmetrical space is examined as Fig. 4 (a). Since each agent is assumed to be able to know its role, the inputs are asymmetric. In this case, the rule that the agent who moves first gives its way, emerged in all the simulation runs in 100 runs. However, in the case of asymmetrical space, the expected rule emerged as shown in Fig. 4 (b).

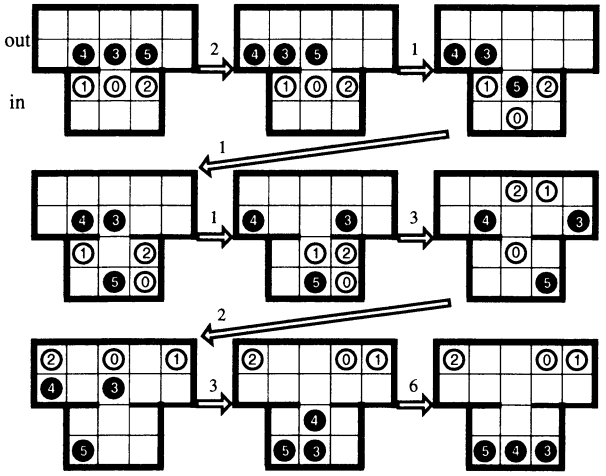
Finally one of the agents is assumed to have a special physical ability, i.e., "strong power". When the opponent exists in front of the agent, no one exists on



(a) in the case of symmetircal space



(b) in the case of asymmetircal space



(c) in the case that a special physical ability is introduced in one agent

Figure 4: Results of the simulations in the case of six agents. It is examined whether the rule emerges that the passengers getting off a train have a priority. The number on the arrows indicates the number of cycles for the transitions. The number in each circle indicates the order of action in one cycle.

the box beyond the front opponent, and no wall exists at the back of the front agent, it can move forward and makes the front opponent go back. Fig. 4 (c) shows the result of the simulation. It can be seen that the agent with "strong power" got on the train at first.

4 Discussion

The authors expect that this model can explain "individuality" and "sociality" in our human society on the micro-level, i.e., individual learning level. We think that the evolution also makes an important role for the emergence. However, the both behavioral characters are defined by observing their actions, and we don't have enough strategy of action when we were born. Then we have dealt with the evolution as one of the factors which influences the learning, and represented it by "physical and mental character".

5 Conclusion

"Individuality" and "sociality" has been introduced, and the model has been proposed in which the both behavioral characters can be obtained by reinforcement learning. It is examined that the model is feasible in some simulations.

Acknowledgments

A part of this research was supported by the Scientific Research Foundation of the Ministry of Education, Science, Sports and Culture of Japan (#10450165) and by Research for the Future Program by The Japan Society for the Promotion of Science (JSPS-RFTF96I00105).

References

[1] Maynard Smith J (1982), Evolution and the Theory of Games, Cambridge Univ. Press.

[2] Zamora J, Millan JR, Murciano A (1998), Learning and Stabilization of Altruistic Behaviors in Multi-Agent Systems by Reciprocity, *Biological Cybernetics*, Vol. 78, No. 3, pp.197-205, 1998

[3] Ota J, Yokogawa Y, Arai T (1996), Study on Asymptotic Strategy-Making in Multiple Autonomous Robot System, *Journal of Robotic Society Japan*, Vol. 14, No. 3, pp. 379-385 (in Japanese)

[4] Smith A (1776), An Inquiry into the Nature and Caufs of the Wealth of Nations

[5] Shibata K, Ito K (1999), Emergence of Communication for Negotiation by a Recurrent Neural Network, *Proc. of ISADS(Int'l Sympo. on Autonomous Decentralized Sys.)'99*, pp.294-301

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

EXPERIMENTS OF A WHEEL TYPE MOBILE ROBOT

Masanori Sugisaka
Artificial Liferobotics Laboratory
Department of Electrical and Electronic Engineering
Oita University
700 Dannoharu, Oita, 870-1192, JAPAN
Tel: +81-975-54-7831 Fax: +81-975-54-7820

E-mail: msugi@cc.oita-u.ac.jp, Home Page http://arob.cc.oita-u.ac.jp/

M. Hara
Department of Electrical and Electronic Engineering
Oita University
E-mail: mhara@cc.oita-u.ac.jp

Abstract

A wheel type mobile robot has been used in various fields. In this paper, we performed the running tests of a mobile robot. The software to avoid obstacles is developed. The experiment results were compared with some control logic by using criterions both the errors from the designed location and smoothness of the running operation.

1. Introduction

Recently, we developed [1] the fuzzy logic controller in order to run the mobile vehicle on the smoothness. However, our results obtained in [1] were for running tests on circular guide line tapes with constant radius. From the experimental results, we see that the fuzzy controller proposed in this paper is robust and has high control performances with respect to both the smooth running and the error from the desired point. We think that we need smaller robot. We developed a wheel type mobile robot.

A wheel type mobile robot has been used in factories or various fields. We performed the running tests of a mobile robot using some control method.

The mobile robot consists of two DC servo motors, one board-computer, and supersonic and photo sensors.

The results obtained using fuzzy controller were compared with the results obtained from another control logic by using criterions both the errors from the designed location and smoothness of the running operation.

2. Mobile Robot [2]

The mobile robot consists of two DC servomotors, one board-computer, PWM controller (for DC motor), photo sensor, supersonic sensors and shock bam per sensor, battery (6V for logic circuit, 6V and 12V for motor), some logic circuit. The mobile robot is driven by two DC motors that are controlled by the one board-computer based on the data obtained from the sensors. Two DC motors are driven by the PWM control.

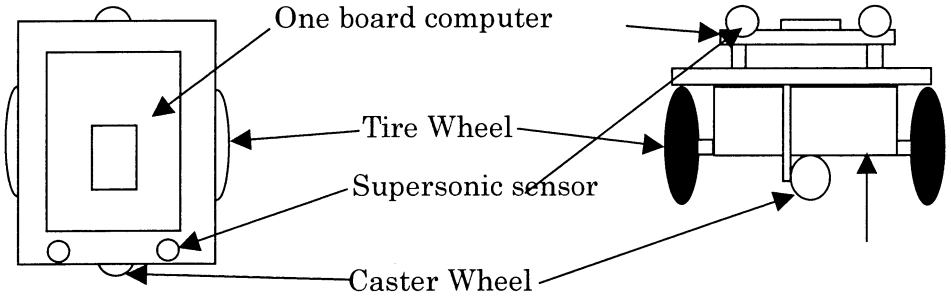


Fig. 1.1 Top View

Fig. 1.2 Front View

Fig.1 Mobile Robot

3. Methods of Experiment

We show the configuration of the mobile robot in Fig.1. As shown in Fig.1 the mobile robot consists of two casters, supersonic sensors, two DC Servo motors and one board computer. In other words, the mobile robot system has two driving wheels and two casters power wheeled steering architecture. DC motor is chosen two types, drived by 6V and 12V.

Mobile robot can sense obstacles using supersonic sensor and photo sensor. Supersonic sensor is set on front top of mobile robot and can sense wide range, but this valid distance is short. And supersonic sensor can measure from mobile robot to obstacle distance.

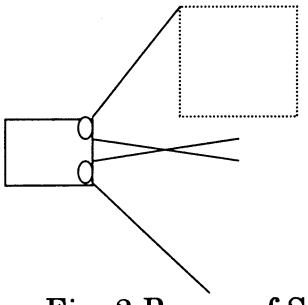


Fig. 2 Range of Sensor

3.1 Running Test

At first, we test to move mobile robot along course ① in Fig.3, where there are no obstacles in this room, for get acknowledgement of this mobile robot by using rule table control. And we experiment course ② or ③ in Fig.3.

Next, we set some obstacles in the room and try to move mobile robot in this environment. Mobile robot can move from start-point to goal-point, but sometimes lost the course and turn around. We think some rule table and more complexity rule are necessary. The result of experiments is not good. We experiment another control method. The results obtained using some control methods were compared with the rule table by using criterions both the errors from the designed location and smoothness of the running operation, in another word smooth moving.

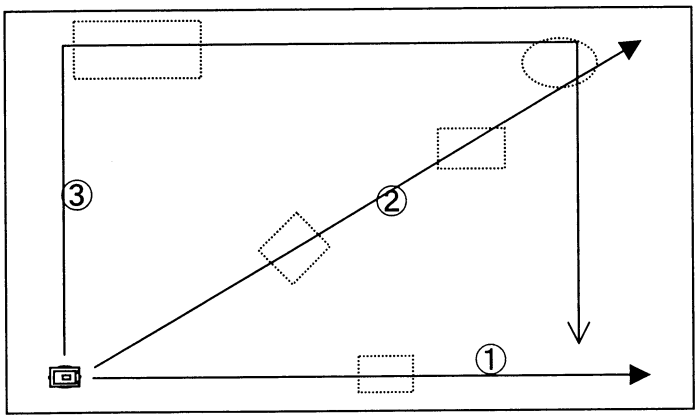


Fig. 3 Experimental Course

3.2 Control Method

DC servomotors are controlled by the one board-computer based on the data obtained from the sensors. Two DC motors are driven by the PWM controller. This PWM Duty ratio is calculated by the methods such as rule table, fuzzy and etc.

Rule Table:

We showed rule in table 1. It is most simple rule. The velocity of motor is this mobile robot has some sensor

Table 1 Most Simple Rule Table

Left sensor signal	Right sensor signal	Left Motor	Right Motor
on	off	up	down
off	on	down	up
on	on	Stop (hold)	Stop (hold)
off	off	Keep (hold)	Keep (hold)

Hold means that if now both sensors are on, left motor speed is hold and right motor speed is zero. It is turning-over.

Fuzzy Control:

Fuzzy logic controller is described by the form of if-then rule such that

If e is A1 and Δe is A2

then Δu is B1

where e is the error of last sensor condition between present sensor condition and Δe is the difference of the error and Δ u is the difference of control value.

The reason why the number of division of regions is few is reduction of the freedom of parameters to be selected for designing fuzzy logic controller. The fuzzy logic control rule is described by if-then form.

The fuzzy reasoning used is min-max method. The output from the fuzzy logic controller is the difference of control Δu_n and the numbers of pulses, ur_n and ul_n per sampling time to the right and left DC motors are calculated from

$$ur_n = ur_{n-1} - Ku \times \Delta u_n,$$

$$ul_n = ul_{n-1} + Ku \times \Delta u_n.$$

respectively. The value of normalizing coefficient Ku = 1.4 is employed from

the experiments for the fuzzy logic controller.

We experiment for use this fuzzy rule and change the rule set. More sophisticated rule is needed.

From the comparison of the results obtained from the experiments [2], [3] on the PI and fuzzy logic controller in the sense of stability and robustness and the fuzzy logic controller is the best among them in order to obtain smooth moving of the mobile robot.

4. Conclusions

In this paper, the running tests of the mobile robot were performed using rule table and the fuzzy controller. If we increase the number of supersonic sensor and photo-sensor, then the control performance using the fuzzy control logic will be increased.

The results obtained using fuzzy controller was compared with the results obtained from PI control logic by using criterions both the errors from the designed location and smoothness of the running operation.

Especially more sophisticated control method such as fuzzy logic control method or neural control method than proportional control method could be employed in order to achieve smooth running.

Reference

[1] M. Sugisaka, M. Hara: "Running test of mobile vehicle", Proceeding of International Workshop on Advanced control (1994), p.439-442.

[2] J. L. Jones, A. M. Flynn: "Mobile robots", A K Peters, Ltd (1993)

[3] M. Sugisaka, M. Hara: "Neuro -Fuzzy control for recognition and tracking system of moving objects ", Proceeding of First Asian Control Conference Vol. 1.1 (1994), p.1091-1098

[4] M. Sugisaka, M. Hara: "Driving control of visual system", Proceeding of the 10th Korea Automatic Control Conference (1995), p.131-134

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Autonomous mobile vehicle

Masanori Sugisaka
○Takuya Adachi

Department of Electrical and Electronic Engineering, Oita University
Oita, 870-1192, Japan
msugi@cc.oita-u.ac.jp
tadachi@cc.oita-u.ac.jp

Abstract

This paper shows the result about the rotation control of DC motor to drive the mobile vehicle, as a first step of the research for the realization of the autonomous mobile vehicle. This autonomous mobile vehicle has one CCD camera driven by stepping motor with encoder, and DC servo motor which drive a rear wheel and DC motor which drives a steering wheel. We show the control methods. This research has used the P control, the PI control and the PID control low to control the speed of the autonomous mobile vehicle.[1],[2],[3]

1. Introduction

Recently, autonomous mobile vehicle is widely used mainly at the factory and hospital in the field of the conveyance work. Different kinds of mobile vehicle will be developed to assist human beings according to the progress of science and technology. Therefore, we carry out our research on mobile vehicle that does not need the human beings help and distinguishes the course automatically, and avoids obstacles, and researches the final goal. The autonomous mobile vehicle is that mobile vehicle moves automatically. If we put the autonomous mobile vehicle into practical use, its use is practical and its work is more correct and more higher efficiency.

2. The control method of the DC motor

2-1. The characteristic of the DC motor

The DC motor is set up into all the products owing to the easiness and the high performance of the control is known. The basic connection and characteristic of the DC motor become like Fig.2-1-1. Because the number of rotations is proportional to the voltage that is fended to the motor, the control of the DC motor is changed by the supply voltage to the motor. There is many kinds of the control methods of DC motor to the stability and

then the desired number of the rotations and the technique, is established. Because the robot and the control equipment is required fairly output torque to motor itself, the motor is large-sized. Consequently, consumption of electric power of the motor itself becomes large. Therefore, the real control of the motor has large output that becomes a point how makes consumption of electric power small.

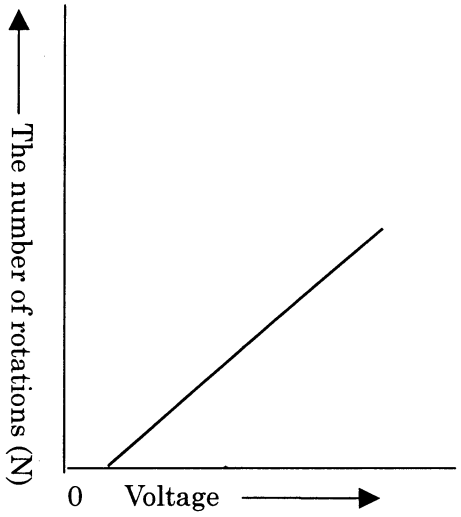


Fig.2-1-1 Characteristic of DC motor

2-2.The drive method of the DC motor

In the case of velocity control, the DC motor can't make an electric current a direct operation variable. Therefore, it supplies the fixed voltage source to the DC motor using a transistor and so on, and using the Pulse Width Modulation method which can change the voltage added to the DC motor that changes velocity of the DC motor.

The velocity of the DC motor is adjusted by ratio with pulse width of ON and OFF to output to the DC motor. This ratio with pulse width is called duety ratio, and Figure 2-2-1 shows time of ON is T_{ON} and time of OFF is T_{OFF}.

$$T_{ON}+T_{OFF}=100 \qquad (1)$$

Duety ratio D is given by

$$D=\frac{T_{ON}}{T_{ON}+T_{OFF}}=\frac{T_{ON}}{100} \qquad (2)$$

From Eq.(1), because T_{OFF} is decided if T_{ON} is decided. Duety ratio D is changed by the Eq.(2) and velocity of the DC motor changes. Value of deuty ratio D is larger, velocity of DC motor becomes faster.

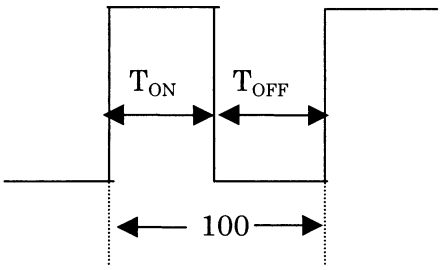


Fig.2-2-1 Duety ratio

3. PID Control

PID control is explained as follow. In the case that noise is not able to be measured, the observation quantity and the control quantity without noise is same quantity.

Then it has pulled from the mark value to the observation quantity shows the desirable value and the actual value of the control quantity discrepancy. This thing is called deviation (or control deviation), the operation quantity is decided by based on this deviation. If the deviation is small, the operation quantity is small too, and if the deviation is large, according to it, the operation quantity is large too, it is suitable. I concludes the proportion clause in the control quantity. This is called the proportional operation. P operation is called from an initial of proportional. If the control object which has self balance that does only proportional control, to mark value and noise of step shape change, the constant deviation has been left. This is called the regularity deviation or off set. This regularity deviation can be removed to include proportional clause to the deviation integral in the control law. This is called integral operation, and PI operation is called from an initial of integral. Because it is purpose to deny the regular deviation, it sometimes says reset operation. Because trend of the deviation fluctuation is reflected by decision of the operation quantity, it contains proportional clause, too, in control law in the derivative of deviation to attempt to improve a control characteristic. This is called derivative operation, PID operation is called from an initial of derivative. It is the foresight operation of a kind. From being proportional to the change percentage with deviation, it is called the rate operation. The control law which contains three pieces operation is PID operation, this is called the three clause control (the three clause operation). The control law of PID control shows a deviation in e,

$$U(t)=K_P\{e(t)+\frac{1}{Ti}\int_0^t e(x)dx+T_D\frac{de(t)}{dt}\}.$$

K_P is proportional gain, T_1 is integral time, T_D is derivation time. Proportional gain is ratio with deviation and operation quantity in case of proportional operation, time that output the proportional operation and the derivation operation to input of same step shape becomes equal is integral time, time that the proportional operation and the derivation operation to input of same straight line shape becomes equal is derivation time.
If deciding to adopt PID control, the design of control system is arranged to decide value of three pieces parameter (the proportional gain, the integral time, the derivation time)

4. The control experiment of the rear wheel of autonomous mobile vehicle

The curving guide line like cosine curve is set in front of the vehicle, and

simulation experiment is performed the DC motor is controlled by P control by in order to decide proportional parameter K. Next, PI control is performed to decide integral parameter T_I by using proportional parameter decided by P control. Finally, PID control is performed to decide derivation parameter T_D by using proportional parameter K and integral parameter T_I decided by PI control. From their data, the parameters of PID control which is the optimal is used for the motor control of the rear wheel part.

5. Consideration of experiment

The purpose of this experiment is rotation number control of DC motor for the rear wheel drive. It adopted three (P, PI, PID) controls. The optimal parameters of PID control from are decided the number of rotation and the mean square error. The graph between the number of rotation and the mean square error based on the experiment data is made and optimal parameter to DC motor was decided from condition and value of graph. The result like Table 1 was obtained. But the welter of the number of rotation and the square error is little seen. This cause is considered that there is noise like influence of friction of wheel axis and gear because output is low.

Table 1 The optimal parameters of PID control

	Pattern 1	Pattern 2
K	3.7	4.1
T_I	12	11
T_D	1.7	0.2

6.Conclusion

As the problem in the future, optimal parameter of DC motor is decided by in outdoors by the small size, the light-weight-size in vehicle. We use an incorporation control of CCD camera and of steering, control of rear wheel part

7. Reference

[1] M.Sugisaka "The system theory and the control", in Japanese Nikkankougyou PP.79-166,1997

[2] N.Suda "PID control", in Japanese Asakura shoten PP.1-19,1992

[3] S.Miyazaki, H.Miyazaki" The applica tion learning of automatic control to learn with the personal computer", in Japanese CQ Shuppansya PP13-26,1992

[4] A Kawanisi "TURBO C beginner's class program", in Japanese Gizyutu hyouronsya PP65-165,(1989)

[5] Y Kurose T Matuo "Turbo C program",in Japanese Syoumei sha PP1-70,(1992)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Studies on 3-D image processing for autonomous vehicle

Masanori Sugisaka and Noriaki Mitsuo

Department of Electrical and Electronic Engineering , Oita University
700 Dannoharu , Oita , 870–1192 Japan
msugi@cc.oita-u.ac.jp
nmitsuo@cc.oita-u.ac.jp

Abstract

In this study, we developed an image measurement system based on CCD camera and three dimensional image processing system for autonomous vehicle using stereo image-based three-dimensional image processing technology which is one kind of three dimensional image processing technology. Stereo image sensing can deal with vision images obtained from two cameras that are effective for distance measuring and image recognizing.

Using two cameras placed lengthwise, our experiments measuring distance of object by difference portrait proved our methods proposed effective.

1 Introduction

Recently the necessities and needs of development of intelligent robot and automatic control self-driving movement vehicle in order to substitute for cooperation with human beings in the field of the welfare and so on are considered now by some researchers all around the world.

When our human beings move here and there in real world, we use our eyes as vision sensors to find, identify objects around ourselves and to avoid collision with them. For robots or self-driving vehicle, such vision sensors are also needed.

So far, the infrared rays and the supersonic waves measured a distance to the object, and the switch judged an obstacle mechanically.

But, we must take the thing of the object that for example the human being and the animal move to, not only the straight line-like distance but also the surrounding environment into consideration with the autonomous vehicle, too.

In this paper, we experimented with "vertical stereo method" as one of the distance measurement methods. This method is based on triangular surveying method. There were a vertical stereo method and a horizontal stereo method in the stereo method but we thought that we had better adopt a vertical stereo for the purpose this time because it makes the level lines which are many at the vehicle correspond compared with the side stereo method which arranges a camera in the direction of the side with the stereo, little extraction leakage by the environment change occurs.

2 The basic principle

2-1 The distance measurement by the stereo view

In "the picture sensing of three dimensions" which measures the position and the shape of the object one three like dimension from the picture, "the sensing of three dimensions by stereo method" which uses the picture which was picked up image from more than one viewpoint is regarded as principles that a human being can catch a distance sense.

At present, distance measurement by the picture is extensively studied using this technique.

This technique is observing an object from the two viewpoints that the mutual positional relation is understood.

We regarded as one of the principle of this technique, "The triangular surveying method" in this study.

This method is based on geometrical property that the triangle form the vertex of which is one on two viewpoints and objects from the visual information can be fixed.

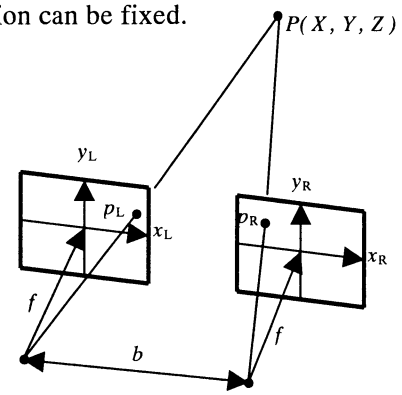


Fig.1 Principle of "The triangular surveying method"

The composition figure of the sensing of three dimensions by stereo method which applied triangular surveying method is shown in Fig.1.
This meets the following three conditions by the camera composition that is in the fact and often used by the stereo view.

1. Optical axis is parallel each other.

2. A line that between optical center both cameras and optical axis cross at right angles each other.

3. Image plane of both cameras are the same plane that cross optical axis at right angles.

(The focal length of both cameras are equal from condition 2 and 3).

Moreover, we put coordinate of object: $p(x, y, z)$, image coordinate of left screen: $p_L(x, y, z)$, and image coordinate of right screen:$p_R(x , y , z)$.

Geometric relation of both projection points yields

$$y_L = y_R = y . \qquad (1)$$

Using above expression, the relation of coordinate of object and coordinate of left-right projection point is

$$X = \frac{bx_L}{x_L - x_R} \qquad (2)$$

$$Y = \frac{by}{x_L - x_R} \qquad (3)$$

$$Z = \frac{bf}{x_L - x_R} \qquad (4)$$

where, f is the focal length of cameras, b is the length of both cameras of optical center we call this base line.

In addition, we often use this expression, $x_L - x_R$, is the gap of projection point both camera's image, parallax. If we use the following expression,

$$d = x_L - x_R \qquad (5)$$

Eq.(4) is transformed as follows.

$$Z = \frac{bf}{d} \qquad (6)$$

The significance of Eq.(6) is what the depth, Z, can be calculated easily if parallax can observe.

To be clear from Eq.(6), It is in the relations of the inverse proportion as for Z and d. Therefore, the parallax becomes small as much as the becoming with far distance the parallax becomes zero if distance is infinity.

The stereo view is the way of trying to fix the three-dimensional position of the object automatically from two sheets of pictures that were taken with the camera to have put on two viewpoints. But as for the fact, a coordinate value isn't directly fixed from two sheets of these pictures. The correspondence between two sheets of pictures must be fixed.

Because, if there is not information that each point of the object that is taken one of the pictures corresponds to which point of the other picture, the actual coordinate value is because it isn't possible to decide.

The condition to search a corresponding point includes epipolar condition This is described below.

2-2 The "Epipolar condition"

Some agreement that occurs from the relation having to do with learning of the several is in the position of the projection point on two sheets of pictures not freely.

When any point in a picture is observed, the position in the space to the point is somewhere of the view line.

In Fig.2, on the picture which was considered from another viewpoint, the projection point is the image of the point which is on the straight line. The straight line passes p_1, making viewpoint e_1 an end point if directing attention to the position on the view line in the space, that is, in one optional p_1 which is deteriorated into picture I_1. If view line e_1p_1 is the straight line that the eyes can see, the image when observing this from e_2 becomes straight-line l_2 on I_2. This is called epipolar-line of point p_1 and also the restriction that the projection point on either side should satisfy is called epipolar restriction. The epipolar line always becomes a pair in the picture on either side and appears.

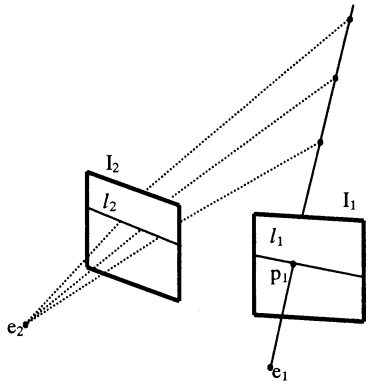


Fig.2 the " epipolar condition"

Because it is good if doing among pairs of the epipolar line when searching a corresponding point, it changes from the two-dimension search to the one-dimension search. This makes computational load little and facilitates the search of the corresponding point.

Actually, when composing a system, optical axis of two cameras becomes parallel like Fig.3. The straight line that joins two viewpoints crosses, being perpendicular to optical axis. We often puts a camera for one coordinate axis of the image plane to become parallel to e_1. Then, a pair of the view line image becomes a simple straight line on the picture. This facilitates to find a corresponding point more. [1]

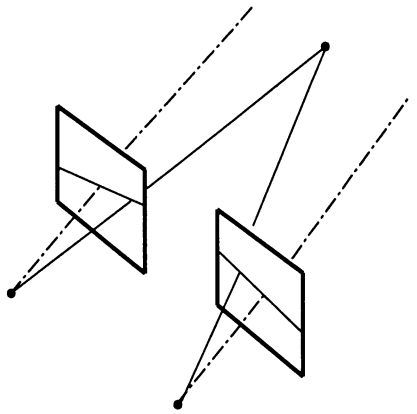


Fig.3 epipolar restriction in the actual system

2-3 The search of the corresponding point

When searching a corresponding point between two pictures, we don't execute a search to all points in the picture. The reason is that: computational load become enormous and finding the correspondence becomes difficult.

Generally, the corresponding point between two screens can be searched only to the point having a characteristic which is different from the neighborhood. Also, it is possible to see observing point in outside world from two viewpoints. This is similar to put the presupposition to fix corresponding point by the picture on either side. However, it becomes difficult especially to find the corresponding points , because the various weather conditions and building arrangement, also moving image [2].

3 Experiments and results

The difference image of two images obtained from two cameras is an important parameter for the distance measurement.

A parallax of both cameras is used by extracting this difference image. We estimate the distance from this parallax, and this is the basic algorithm of the distance measurement.

This is verified from this experiment. In this experiment, a square figure 100 x 100mm is put on the wall and a camera is set on the position of 2m from the wall. We use the vertical stereo as a measurement device. The difference image is extracted two sheets of images that obtained from two cameras with GPB-K.

Taking images is repeated by changing the distance from the wall distance by each 0.5m. A proofreading curve with parallax and the distance is obtained using the fact that a distance is in proportion to parallax of the difference image. The Table 1 shows the specifications of cameras used to take images. The result of the experiment is shown in Fig.4 [3]

Table 1 A condition in the experiment

Item	values and condition
arrangement	Vertical
length of base line	50mm
Lens	16mm 1:1.4
Focus	Infinite
Aperture	8

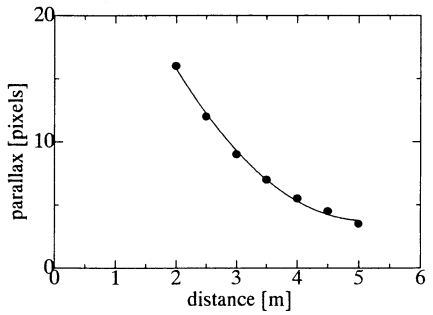


Fig.4 Relation with the distance and parallax

4 conclusion

Our experiment was verified about the validity of image measuring with two eyes stereophonic form that is a pre-stage of the three-dimensional image processing for a car. It is found out what could be done in the precision for measuring distance when a camera was fixed on a car and a camera was movable . The search of the point which copes with distortion, a shadow, and so on appear in the image from the position to take pictures, the degree of the incidence of the light, and so on becomes difficult. The method such as neural network and the genetic algorithm will be used to three-dimensional image processing for a car in future. [4]

References

[1] T.Okutomi "Stereo Image-Based 3-D Sensing", Tutorial lecture meeting basis course textbook of the 4th Symposium on Sensing via Image Information, pages 25-28, Jun.1998.

[2] T.Agui, T.Nagao, "Computer Image Processing and Recognition" Shoukoudou, pages151, Nov.1992.

[3] N.Mitsuo, "Foundation research about three-dimensional image measuring for carrying a car." Graduation thesis of Oita University, pages 1-40, Feb.1999.

[4] M.Sugisaka "The system theory and the control", pages 158-160,1997

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

The steering control of the mobile vehicle

M.Sugisaka [1] and S.Seki [2]

[1] Department of Electrical and Electronic Engineering

[2] Graduate School of Engineering

Oita University
700 Dannoharu, Oita 870-1192, JAPAN
E-mail: msugi@cc.oita-u.ac.jp seki@cc.oita-u.ac.jp

Abstract

In this paper, we report the results about the steering control of the mobile vehicle , as a first step of the research for the realization of the mobile vehicle with the artificial brain.

1. Introduction

Various mobile vehicles (MVs) will be developed more and more with demand in future too and the elimination of manpower. Recently, the technique about MV makes progress steadily. Now transmitters to guide MV are buried in the ground and a wall side. MV calculates a course by many sensors equipped on a body and controls itself. But there are problems to be solved in this method. One of them is that it requires a great labor to establish the environment. But this problem will be solved by MV, understanding the situation of circumference. MV is not required to establish a course beforehand so that the action of automatic running is possible. When an obstacle exists, MV evades it and can arrive at a destination. We carried out fundamental experiments to autonomous running of MV in this studies. The position of MV is measured using the vision system in the MV. Our purpose is that MV runs smoothly along guideline tape.

2. Mobile Vehicle

Fig.1 shows specification of MV. CCD camera is installed in front of the body because MV judges course from the situation in front and runs. Also, it is driven by a rear wheel using the DC motor. The radius of the wheel is 13 cm. When the motor turns on, MV moves by about 2.7cm.

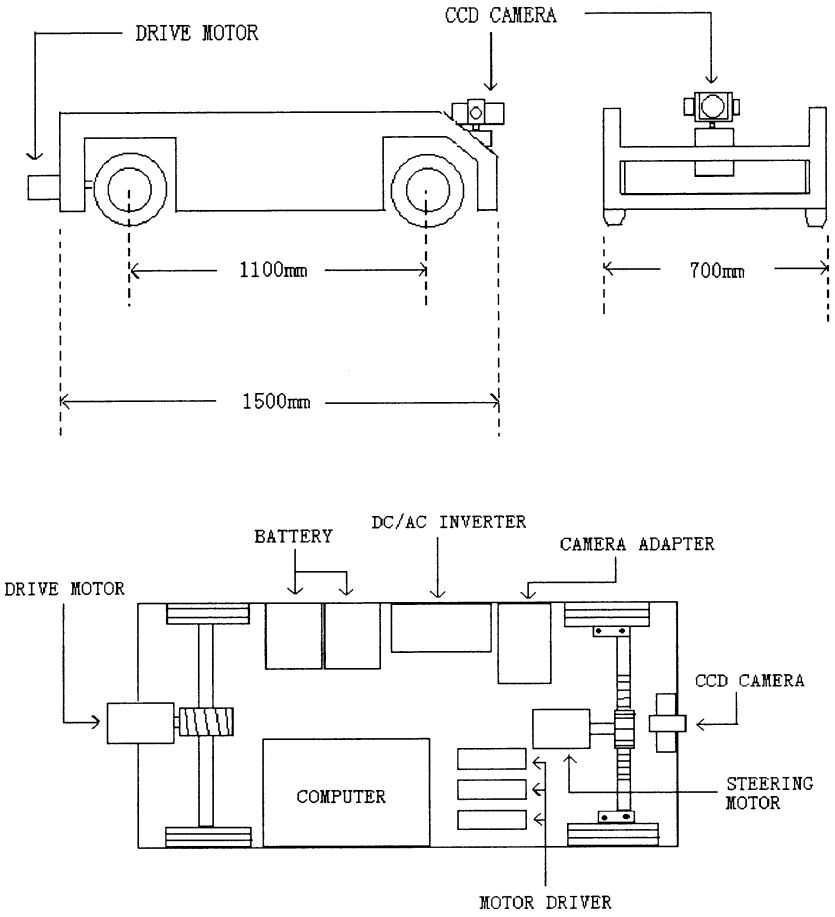


Fig.1 Specification of MV

3. Image Processing

The image taken by the CCD camera is changed into the digital data of R, G, B. R, G, B are independent vector each other which shows red, green, blue, respectively. The optional color can be shown with this ingredient with three primary colors. These ingredients R, G, B are called a three stimulation value. If showing R, G, B as Fig.2, it is possible to make vector C correspond to this point. The size of vector C shows brightness. Then, the direction shows chromaticity. For chromaticity shown by Fig.2, it is shown by the coordinate in the point of intersection P of vector C and the plane which is decided in R+G+B=1. It is possible to show these coordinates r, g, b by the next equation.

$$r = R/(R+G+B)$$

$$g = G/(R+G+B)$$

$$b = B/(R+G+B)$$

Hue which shows the kind of the color and saturation which shows to do the thick of the color are defined. Hue with the color which corresponds to point P in Fig.3 is shown by corner θ in segments WR and WQ to make.

$$\theta_1 = \cos^{-1}\frac{2r-g-b}{\sqrt{6}\left[(r-1/3)^2+(g-1/3)^2+(b-1/3)^2\right]}$$

$$\theta = \begin{cases} \theta_1 & g \geq b \\ 2\pi - \theta_1 & g < b \end{cases}$$

Saturation S with the color which corresponds to point P is given in WP/WQ. Therefore, we obtain.

$$S = 1 - 3\min(r, g, b)$$

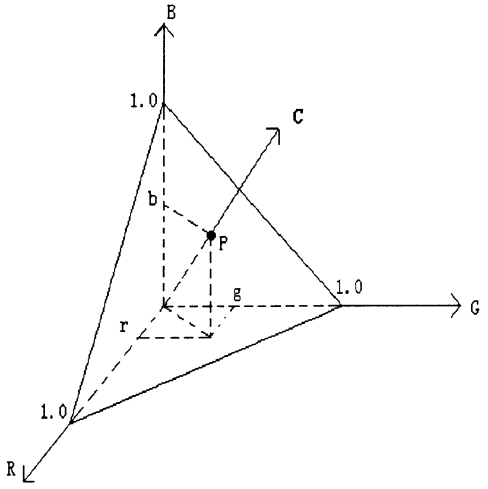


Fig.2 Color Indication on 3D Cartesian Coordinates

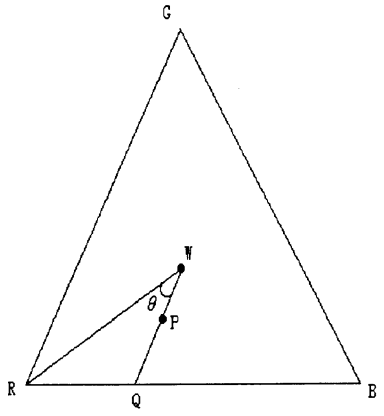


Fig.3 Hue and Saturation

4. Experiment

In the experiment MV runs actually and performs the acquisition of the image, the computation of the center of gravity, and the operation of the steering.

5. Control Methods

5.1 The proportional control

The number to have multiplied a proportional number by the deviation from the center of the body to the center of gravity by becomes operation quantity.
Therefore, a proportional number was changed and the conduct of MV was examined.

See Fig.4.

5.2 The proportional control to have considered the inclination of MV to the guide line tape

The difference with the deviation of the part above and the deviation of the part below becomes inclination degrees. It puts weight to the degrees. In addition to the quantity to operate them, it changes weight and it examined the move of MV.

See Fig.5.

6. Conclusion

In case of experimentally this time, we find a deviation from the center of the body to the center of gravity with goal value. We did the experiment which decides the quantity to operate a steering using the value.

Next, we considered the inclination of MV to the guide line tape and did the experiment which decides the quantity to operate a steering using the deviation which put weight to the inclination.

In the future, we look for the course of the guide line tape using interpolation
based on the image of the CCD camera and plan to experiment in the way of operating a steering using the course.

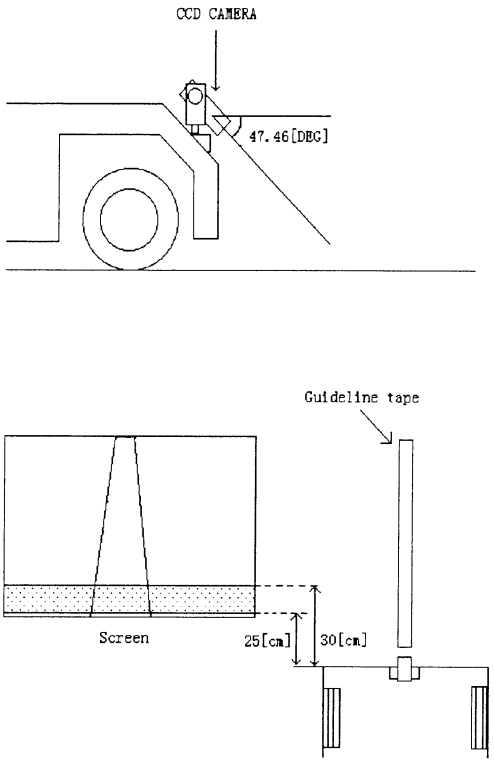


Fig.4 Gradient of CCD Camera and View of MV

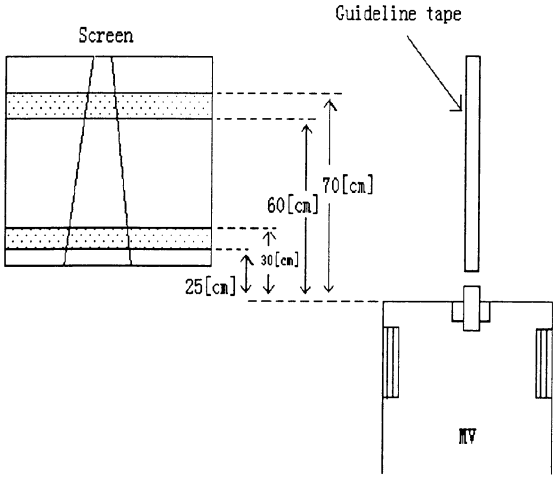


Fig.5 View of MV

7. References

[1] M.Sugisaka "The system theory and the control" Nikkankougyou shinbunsya (1997)

[2] H.Tanaka "The image processing application technique" Kougyou chousakai pp.85-106 (1989)

[3] K.Tezuka. T.Kitabashi. H.Ogawa. " The digital image processing engineering" Nittukan kougyou shinbunsya pp.11-15 pp.121-123 (1985)

[4] T.Nagao " The processing and the recognition of the picture" Syoukoudou pp.176-200 (1992)

[5] N.Kojima "The basis numerical value computation" Tokaidaigakusyupankai pp.105-132 (1989)

[6] M. Sugisaka, Xin Wang, and Ju-Jung Lee, "Intelligent control strategy for a mobile vehicle" Applied Mathematics and Computation, vol.91, No.1, pp.91-98(1998)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Capturing a Moving Object using an Active Camera mounted on a Mobile Robot

Jin Woo Park
Dept. of Electronics Eng.
Pusan Nation University
Pusan, 609-735, Korea

Jae Han Park
Dept. of Electronics Eng.
Pusan Nation University
Pusan, 609-735, Korea

{jwpark, hans1024}@home.pusan.ac.kr

Hwa Ra Hur
Dept. of Electronics Eng.
Pusan Nation University
Pusan, 609-735, Korea

Jang Myung Lee
Dept. of Electronics Eng.
Pusan Nation University
Pusan, 609-735, Korea

{hrhur, jmlee}@hyowon.pusan.ac.kr

Kiyoharu Tagawa
Dept. of Electrical and Electronics Eng.
Kobe University
Kobe, 657-8501, Japan
tagawa@eedept.kobe-u.ac.jp

Hiromassa Haneda
Dept. of Electrical and Electronics Eng.
Kobe University
Kobe, 657-8501, Japan
haneda@eedept.kobe-u.ac.jp

Abstract

In this paper, we propose a method of tracking and capturing a moving object. The position of moving object is acquired by the relation between image information from a 2-DOF active camera mounted on mobile robot. Forwarding direction and rotational angular velocity of the moving object are estimated by state estimator. We use a kalman filter as state estimator for taking characteristics of robustness on noises and uncertainties included in input data. After estimating the trajectory of the target object from information that is estimated by the kalman filter, we decide the optimal trajectory and motion planning of the mobile robot for capturing target object within the shortest time. The effectiveness of proposed method is demonstrated by the simulations.

Keywords : Autonomus mobile robot, Kalman filter
Tracking & capturing, Active camera system

1. Introduction

A mobile robot has many application fields because of its high workability. Especially, it is definitely necessary for the tasks that are difficult and dangerous for man to perform. Many people have interested in the mobile robot. However, most of them are aiming at successful navigation, that is, focusing on reaching at a fixed target point safely. However, if a mobile robot is working under the water or in the space, the target object may move freely. Therefore, ability of processing moving targets is necessary for the mobile robot. Also, if an active camera system is applied to tracking a moving object and navigation, there will be many advantages. An active camera system capable of panning and tilting can automatically calibrate itself and can keep track of an interested object for longer time interval without the mobile robot movement[1].

In this paper, we propose a method of tracking and capturing a moving object. This paper is organized as follows. In the section 2, we discuss how to establish the model of an active camera. Section 3, 4 deal with the problem of trajectory estimation of a moving object and motion planning for capturing a moving object respectively. In the section 5, Simulation result presents the advantage of proposed method. Section 6 presents conclusions drawn from this work.

2. Active camera system

We derive some equations on the image processing for the mobile robot. We begin with considering kinematics for actuator of active camera.

2.1 Kinematics for actuator of active camera system

In our active camera system, It has the ability of panning and tilting as shown in Fig. 1. We define the position and posture of camera about the base frame. According to the Denabit-Hatenberg convention, we can obtain the homogeneous matrix after establishing the following coordinate system and representing the parameter as shown in Table 1 and Eq. 1 respectively.

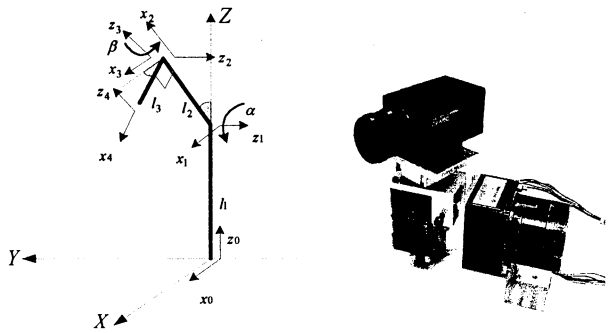


Fig. 1 The two degrees of freedom of the camera planform (Left) and what it looks like (Right)

$${}^{0}H_4 = {}^{0}H_1 \cdot {}^{1}H_2 \cdot {}^{2}H_3 \cdot {}^{3}H_4 = \quad (1)$$

$$\begin{bmatrix} \cos(\alpha)\cos(\beta) & -\cos(\alpha)\sin(\beta) & \sin(\alpha) & l_2\sin(\alpha)+l_3\cos(\alpha)\cos(\beta) \\ \sin(\beta) & \cos(\beta) & 0 & l_s\sin(\beta) \\ -\sin(\alpha)\cos(\beta) & \sin(\alpha)\cos(\beta) & \cos(\alpha) & l_1+l_2\cos(\alpha)-l_3\sin(\alpha)\cos(\beta) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The position of camera by homogeneous matrix is presented as follows

$$\begin{aligned} x_{ccd} &= l_2\sin(\alpha)+l_3\cos(\alpha)\cos(\beta) \\ y_{ccd} &= l_3\sin(\beta) \\ z_{ccd} &= l_1+l_2\cos(\alpha)-l_3\sin(\alpha)\cos(\beta) \end{aligned} \quad (2)$$

It can be obtained a tilted angle and rotated angle by panning and tilting of camera as shown in Eq. 3. θ_T is angle between vector x_4 transformed to value of absolute coordinate system and base frame X-Y coordinate after. θ_R is angle between vector y_4 transformed to value of absolute coordinate system and base frame X-Y coordinate.

$$\theta_T = \tan^{-1}\left(\frac{\sin(\alpha)\cos(\beta)}{\sqrt{\cos^2(\alpha)\cos^2(\beta)+\sin^2(\beta)}}\right)$$

$$\theta_R = \tan^{-1}\left(\frac{\sin(\alpha)\sin(\beta)}{\sqrt{\cos^2(\alpha)\sin^2(\beta)+\cos^2(\beta)}}\right) \quad (3)$$

2.2 Relation between camera and real coordinate

Now that the position where the obstacle meets the bottom can be detected, we can estimate the obstacle position kinematically from the tilt angle and rotation angle of the camera mounted on the mobile robot. Those results are in Eq. 4-6.

$$\begin{bmatrix} j' \\ k' \end{bmatrix} = \begin{bmatrix} \cos(\theta_R) & -\sin(\theta_R) \\ \sin(\theta_R) & \cos(\theta_R) \end{bmatrix} \begin{bmatrix} j-\frac{P_x}{2} \\ k-\frac{P_y}{2} \end{bmatrix} \quad (4)$$

$$\hat{r}_0 = z_{ccd}\cdot\cot(\theta_T - \frac{k'}{p_y}\theta_{ry}) \quad \hat{\theta}_0 = \frac{j'}{p_x}\theta_{rx} \quad (5)$$

$$\begin{aligned} \hat{x}_0 &= r_{ccd}\cdot\cos(\beta)+\hat{r}_0\cdot\cos(\beta+\hat{\theta}_0) \\ \hat{y}_0 &= r_{ccd}\cdot\sin(\beta)+\hat{r}_0\cdot\sin(\beta+\hat{\theta}_0) \\ (r_{ccd} &= \sqrt{x_{ccd}^2+y_{ccd}^2}) \end{aligned} \quad (6)$$

where j', k' are the rotated image by moving of camera, $\hat{r}_0$, $\hat{\theta}_0$ is the estimated values of distance, direction respectively and $\hat{x}_0, \hat{y}_0$ is x, y position of obstacle based on the coordinate of a mobile robot.

2.3 Inverse kinematics for placing the center of image to the desired position

In case of using active camera, we can obtain visual information of area that we want to know but it is required inverse kinematics to obtain information for desired area. Eq.7 is the inverse kinematics equations that describe the relation to place the center of image to the desired position.

$$\beta_d = \tan^{-1}\left(\frac{y_d}{x_d}\right)$$

$$\alpha_d = \cos^{-1}\left(\frac{-l_1l_2+\sqrt{l_1^2l_2^2-(l_1^2+r_d^2)(l_1^2-r_d^2)}}{(l_1^2+r_d^2)}\right)\left(\frac{1}{\sin(\beta_d)}\right) \quad (7)$$

where α_d, and β_d are the attitude of actuator respectively, x_d and y_d are the desired position that we want to place. r_d is $\sqrt{x_d^2+y_d^2}$ and link parameters l_1, l_2 of camera actuator respectively.

3. Trajectory estimation of moving object

3.1 Modeling of moving object

If the target's velocity and acceleration estimates has been obtained, the prediction of the target's future position ($\hat{P}_{fx}, \hat{P}_{fy}$) is accomplishes via the Eq. 8[2].

$$\begin{aligned} \hat{P}_{fx} &= \hat{P}_x+\hat{V}_x\delta t+\tfrac{1}{2}\hat{A}_x\delta t^2 \\ \hat{P}_{fy} &= \hat{P}_y+\hat{V}_y\delta t+\tfrac{1}{2}\hat{A}_y\delta^2 \end{aligned} \quad (8)$$

where, δt is the sampling time $(\hat{P}_x,\hat{P}_y),(\hat{V}_x,\hat{V}_y)$ $(\hat{A}_x,\hat{A}_y)$ are the current cartesian coordinate estimates of the target's position, velocity, and acceleration, respectively. In the X-Y coordinate, Movement of moving object can be divided forwarding velocity element and angular velocity element as follows[3]

$$\begin{aligned} \delta x_{k+\delta,k} &= v_k\delta t\cos(\theta_k+\tfrac{1}{2}\omega_k\delta t) \\ &\approx v_k\cos(\theta_k)\delta t-\tfrac{1}{2}\omega_k v_k\sin(\theta_k)\delta t^2 \\ \delta y_{k+\delta,k} &= v_k\delta t\sin(\theta_k+\tfrac{1}{2}\omega_k\delta t) \\ &\approx v_k\sin(\theta_k)\delta t-\tfrac{1}{2}\omega_k v_k\cos(\theta_k)\delta t^2 \\ \delta\theta_{k+\delta t,k} &= \omega_k\delta t \\ \delta v_{k+\delta t,k} &= \xi_v \\ \delta\omega_{k+\delta t,k} &= \xi_\omega \end{aligned} \quad (9)$$

where, ξ_v, ξ_ω are the variation of velocity and angulare velocity respectively. Using Eq.9, we can obtain the state transition matrix as follows

$$\begin{aligned} x_k &= \Phi_{k,k-1}x_{k-1}+w_{k-1} \\ z_k &= H_kx_k+v_k \end{aligned}$$

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \\ v_k \\ \omega_k \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T\cos(\theta_{k-1}) & -\frac{1}{2}v_{k-1}T^2\sin(\theta_{k-1}) \\ 0 & 1 & 0 & T\sin(\theta_{k-1}) & -\frac{1}{2}v_{k-1}T^2\cos(\theta_{k-1}) \\ 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \\ v_{k-1} \\ \omega_{k-1} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \xi_{vk} \\ \xi_{\omega k} \end{bmatrix}$$

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \qquad v_k = \begin{bmatrix} \gamma_x \\ \gamma_y \end{bmatrix} \tag{10}$$

where, Φ_k is called the state transition matrix, w_k is a vector representing the process noise, z_k is the measurement vector, H_k is a relating the measurement to the state vector, v_k is the measurement error.

3.2 State estimate of moving object by Kalman filter

It need to be covariance matrix of estimated error for getting the filter gain. Projected estimate of the covariance matrix of estimate error is presented as

$$P_k^{'} = \Phi_{k,k-1}P_{k-1}\Phi_{k,k-1}^T + Q_{k-1} \tag{11}$$

The optimal filter gain K_k that minimizes the errors associated with the updated estimate is

$$K_k = P_k^{'}H_k^T[H_kP_k^{'}H_k^T + R_k]^{-1} \tag{12}$$

And the estimete of the state vector $\hat{x}_k$ from the measurement z_k is expressed as

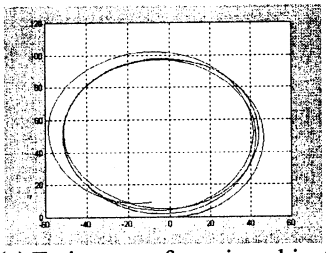
$$\hat{x}_k = \Phi_{k,k-1}\hat{x}_{k-1} + K_k[z_k - H_k\Phi_{k,k-1}\hat{x}_{k-1}] \tag{13}$$

$\hat{x}_k$, therefore, updated based on the new values provided by z_k. The covariance of the error in the estimate P_k can be updated as follows[4];

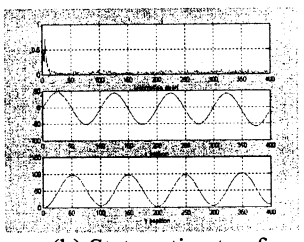
$$P_k = P_k^{'} - K_kH_kP_k^{'} \tag{14}$$

Update time to $k+1$ and return to step Eq. 11

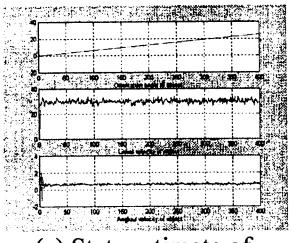
Fig.2 shows example of state estimation using the kalman filter. We simulated assuming measurement, forwarding velocity and rotational angular velocity as gaussian random noise of variance 2, 3 and 0.1 respectively



(a) Trajectory of moving object



(b) State estimate of $|error,|, x_k, y_k$



(c) State estimate of θ_k, v_k, ω_k

Fig. 2 Simulation of state estimate using Kalman filter

In the Fig. 2, We can see that the kalman filter estimated each state of system under existing noise.

3.3 Trajectory estimate of moving object

We can estimate states of moving object if initial state and input are given at the state transition model. It can also be estimated next inputs by estimating forwarding velocity and angular velocity of moving object from state estimator using the kalman filter. In this paper, we get forwarding velocity&acceleration and rotational angular velocity&acceleration and then, approximate them to the equation of the first degree like as follows;

$$\begin{aligned} v_{k+n} &= \hat{v}_k + \hat{a}_{vk}nT \\ \omega_{k+n} &= \hat{v}_k + \hat{a}_{\omega k}nT \end{aligned} \tag{15}$$

In the Fig 2(b), The result has noise although it is applied the kalman filter because it is dynamic system. We, therefore, use least square estimate method that it has characteristics of insensitive against noise.

$$\hat{\theta} = (A^TA)^{-1}A^Ty$$

$$\hat{\theta} = \begin{bmatrix} \hat{v}_k & \hat{\omega}_k \\ \hat{a}_{lk} & \hat{a}_{wk} \end{bmatrix}, \ \mathrm{A} = \begin{bmatrix} 1 & -T \\ 1 & -2T \\ : & : \\ 1 & -mT \end{bmatrix}, \ \mathrm{y} = \begin{bmatrix} v_{k-1} & \omega_{k-1} \\ v_{k-2} & \omega_{k-1} \\ : & : \\ v_{k-m} & \omega_{k-1} \end{bmatrix} \tag{16}$$

From estimated input and the state transition model, Trajectory of moving object can be estimated as follows

$$\hat{x}_{k+m} = x_k + \sum_{h=0}^{m} v(h)\cos[\theta(h)]T, \ \hat{y}_{k+m} = y_k + \sum_{h=0}^{m} v(h)\sin[\theta(h)]T$$

$$v(h) = \hat{v}_k + \hat{a}_{lk}ht, \ \theta(\mathrm{h}) = \hat{\theta}_\mathrm{k} + \hat{\omega}_k hT + \frac{1}{2}\hat{a}_{\omega k}hT^2 \tag{17}$$

4. Motion planning for capturing

In the problem of capturing a moving object, we have to control a mobile robot by considering relation between the position of mobile robot and the position of moving object because the moving object continue to move. Fig. 3 shows motion planning of the mobile robot for capturing the moving object.

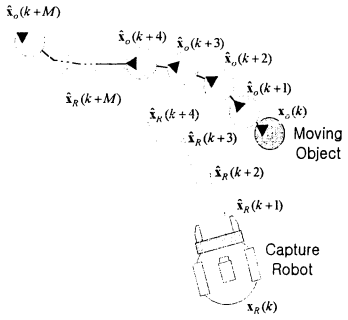


Fig. 3 Estimation of trajectory for capturing

In the Fig. 3, the mobile robot estimates position of the moving object within 1-M sampling time and select the shortest distance from itself to the moving object as like Eq. 8

$$\min_{m=1-M} \left\| \hat{x}_0(k+m) - \hat{x}_R(k+m) \right\| \qquad (18)$$

The position of moving object in the Cartesian coordinate system is acquired by the relation between image information. Forwarding direction and rotational angular velocity of the moving object are estimated by state estimator. We use the kalman filter as state estimator for taking characteristics of robustness on noises and uncertainties included in input data. After estimating the trajectory of the target object, we decide the optimal trajectory and motion planning of the mobile robot for capturing a target object within the shortest time. Fig.4 shows the overall structure for control of mobile robot for capturing a target object.

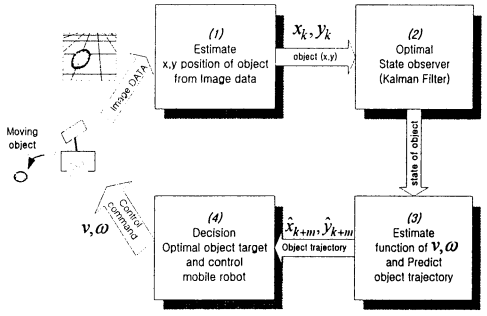


Fig. 4 Control of a mobile robot for capturing

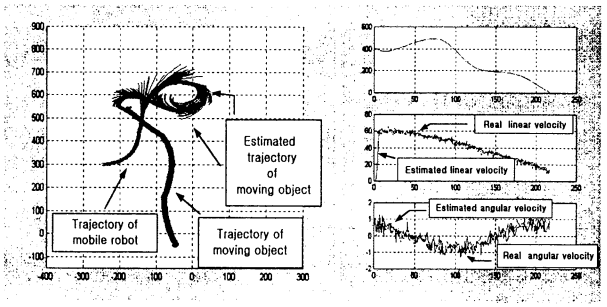
5. Simulations

To demonstate how to use the proposed method, we present example. We assume that limit velocity of mobile robot is 30cm/sec and the mobile robot is higher than the moving object. Initial point of mobile robot and the moving object are (-50, -50), (-250, 300) respectively. Velocity and angular velocity of moving object is as follows;

$$\begin{aligned} v(k) &= 30(\cos(0.01k)+1)+\xi_v[cm/\sec] \\ \omega(k) &= 0.7\sin(0.03k+\pi/1.5)+\xi_\omega[rad/\sec] \end{aligned} \qquad (19)$$

and forwarding direction and rotional angular velocity are gaussian random variable with variance 2, 0.1 repectrively. Fig. 5 is the result of simulation.

Fig. 5(b) are the distance between the mobile robot and the moving object, the error between estimated velocity and real velocity, the error of estimated angular velocity and real angular velocity respectively. it can be seen that although error of estimated information is high at the first time, the error is convergence in a minute.



(a) Trajectory (b) Estimated state

Fig. 5 Result of simulation

6. Conclusions

In this paper, we proposed the method of tracking and capturing a moving object. Proposed method is divided as follows.

1. Position estimation of a target object by considering kinematics relation of active camera
2. Movement estimation of the target object using the kalman filter for tracking
3. Motion planning of a mobile robot for capturing the target object with the estimated trajectory within the shortest time

We are in process of experiment for the proposed algorithm using the mobile robot ZIRO$_2$[5]. Then, we are planning to solve the problems arising from experiment, that is, Real-time image processing, Uncertainties on actuator of active camera and the status of mobile robot.

References

[1] K. Daniilidis, C. Krauss, "Real-Time Traking of Moving Objects with an Active Camera," *Real-time imageing, Academic Press Limited*, 1998

[2] Russell F. Berg, "Estimation and Prediction for Manueuvering Target Trajectories," *IEEE Trans. on Automatic Control*, vol. AC-38, no.3, Mar., 1983.

[3] Steven M. Lavalle and Rajeer Sharma, "On Motion Planning in Changing Partially Predictable Environments," *The Int. Journal of Robotics Research*, vol. 16, no. 6, pp. 705-805, Dec., 1997.

[4] H. W. Sorenson, "Kalman Filtering Techniques," *Advances in Control Systems Theory and Applications*, vol.3, 1996, pp.219-292

[5] J. W. Park, J. M. Lee, "Robust Map Building and Navigation for a Mobile Robot using Active Camera," *Proc. of ICMT*, pp.99-104, Oct. 1999.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Development of Mobile Robot for Autonomous Navigation on Line of Welding Joint by Using Genetic Algorithm

Satoshi Kushida*1 , Tohru Haseda*1 , Mikio Kuge*2 and Dong-Ying Ju*3

*1 *Department of Materials Science and Engineering, Saitama Institute of Technology, Fusaiji 1690, Okabe, Saitama 369-0293, Japan.Email:*mat98004@sit.ac.jp

*2 *ASPECTS INC., Hikashi-Kasai, 6-2-3,Edogawa-Ku,Tokyo,134-0084,Japan*

*3 *Department of Mechanical Engineering, Saitama Institute of Technology Fusaiji 1690, Okabe, Saitama 369-0293, Japan. Email:* ju@sit.ac.jp

ABSTRACT

In this paper, an intelligent control of a mobile robot in order to inspect damage in welding joint of various steel structures is presented. In the mobile robot, a drive method with caterpillar of equipped stronger magnet coil is employed to navigate on wall of a welding structure. And, a few of photo sensors and a CCD camera are equipped in the robot for recognizing joint line of welding and inspecting welding surface. Here, a genetic algorithm for predicting damage information in the joint line of welding is proposed, and it is possible that recognized map and photograph on the joint line of welding be reproduced on the screen for remote control. Depending on the experiments, reliability of the genetic algorithm is confirmed in the reality of the welding.

Keywords: Inspection of welding, Mobile robot, Genetic algorithm, Intelligent control system.

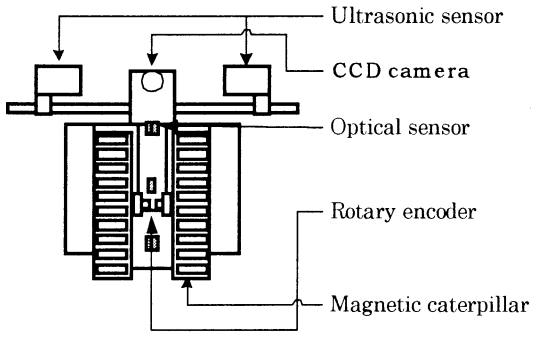
1. INTRODUCTION

In prediction and inspection of damage in welding joint of the building, bridge and lager-scale structures, the flaw detector by ultrasonic wave often is used into the inspection of engineering. However, in order to raise inspecting efficiency, it is necessary to develop a mobile robot for autonomous inspection. The ability to guarantee the feasibility of a mobile robot during autonomous navigation for inspection on a joint line of welding is important to propose an intelligent controller or algorithm. Especially, when the inspection is completed to follow the local recognition of a welding line for the task of autonomous navigation, the safe position of the mobile robot has to be satisfied, while the orientation can be arbitrary. On the other hand, with an autonomous navigation and inspection on welding joint, an intelligent control is important exactly to recognize its own position for the robot and to utilize the inspection and on-line control on the welding joint line. Recently, a lot of studies for proposing intelligent control combined with fuzzy and genetic algorithms are carried out, for example references [1]-[6]. However, them applications in engineering problems remain to be solved. In this connection, an intelligent control of a mobile robot in order to inspect damage in welding joint of various steel structures is presented in this paper. In development of the mobile robot, a drive method with caterpillar of equipped stronger magnet coil is employed to navigate on wall of a welding structure. And, a few of photo sensors and a CCD camera are equipped in the robot for recognizing joint line of welding and inspecting welding surface. Here, a genetic algorithm for predicting damage information in the joint line of welding is proposed, and it is possible that recognized map and photograph on the joint line of welding are reproduced on the screen for remote control. Depending on the experiments, reliability of the genetic algorithm is confirmed in the reality of the welding.

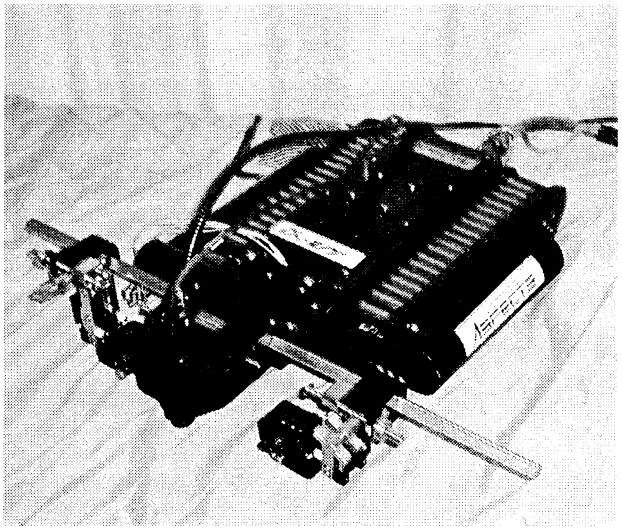
2. SYSTEM OF MOBILE ROBOT

In the research work, a mobile robot (only with A4 size of a paper) developed by ASPECT Corporation is employed. The robot is provided with two caterpillars in which equipped stronger magnet coil for navigation on vertical wall of a welding structure. And, equipped five photo sensors are used to recognize line of welding joint. As improvement of the robot shown in Fig.1, a CCD camera is equipped in the robot for inspecting welding surface. On the other hand, in order to increase ability of control and utilize intelligent algorithm, a Z80 CPU board in original system of the robot is replaced to a V25

CPU board. The CPU board is employed to control navigation of the mobile robot and to judge position of welding joint line depending on the optical sensors. The ultrasonic sensor of the robot is jointed to a flaw detector by ultrasonic wave. The CCD camera is jointed to other PC computer for inspecting the damage on surface of the welding joint.



(a) System of inspection robot



(b) Photogragh of the robot

Fig. 1 Synoptic representation of the inspecting robot.

3. AUTONOMOUS NAVIGATION BASED ON GENETIC ALGORITHM

In autonomous inspecting of welding joint, the ability to guarantee the feasibility of the mobile robot during autonomous navigation for inspection on a joint line of welding is very important to propose an intelligent controller or algorithm. In order to utilize the objects, a genetic algorithm is applied into navigation of the mobile robot.

3.1 Genetic encoding and evolving of gene

In design of the genetic algorithm, the count number in which optical sensors recognize the welding joint is defined as a generated gene. However, we deal with the counter number recognized from the two sensors at front and back of the robot as deference gene individuals BL and FL, respectively. On the other hand, when left (or right) sensor at front and right (or left) sensor at back of the robot at the same time recognize the welding joint, the gene for describing round of the robot is defined by TL. Table 1 shows a sample of gene group. Here, the number in Table 1 denotes the information of these genes, so that evolving of the genes due to crossover can be obtained. However, when the number is five, the present control condition will be remained to variation of information at next crossover step as a present control level PCL.

Table 1 Group of gene

BL	FL	TL
9	9	1
7	7	3
6	6	**PCL=5**
PCL=5	**PCL=5**	7
3	3	9

3.2 Evaluation for the robotic navigation

In order to reflect randomness of the fitness λ_f and λ_b in selected gene individual, a scaling method of index type is introduced into the algorithm, i.e., the fitness at front and back of the robot can be expressed as follows as

$$\lambda_f = (BL - PCL) + (PCL - TL) \quad (1)$$

$$\lambda_b = (FL - PCL) + (PCL - TL) \quad (2)$$

Here, the gene individual with large probability will be selected to next step for crossover operation. In the paper, random crossover based on one-point method[6]-[7] is used as shown in Fig. 2.

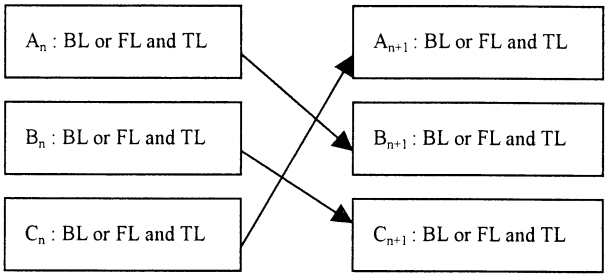


Fig.2 Crossover of gene group.

3.3 Selection

Depending on computed fitness, we can select optimum gene group to decide the control parameter for next step of navigation. The selection algorithm is given by

$$S(0) = 0 \tag{3}$$

and $$S(i) = \sum_{j=1} \lambda(X_j) \quad i = 1,2,\cdots,p \tag{4}$$

where p is the population-size.

3.4 GA operation

We propose a special framework shown in Fig.3 to integrate the intelligent algorithm as a control method for inspecting damage on welding surface by using autonomous navigation. Depending on the flowchart, we can decide increase of distance and direction in navigation of the mobile robot.

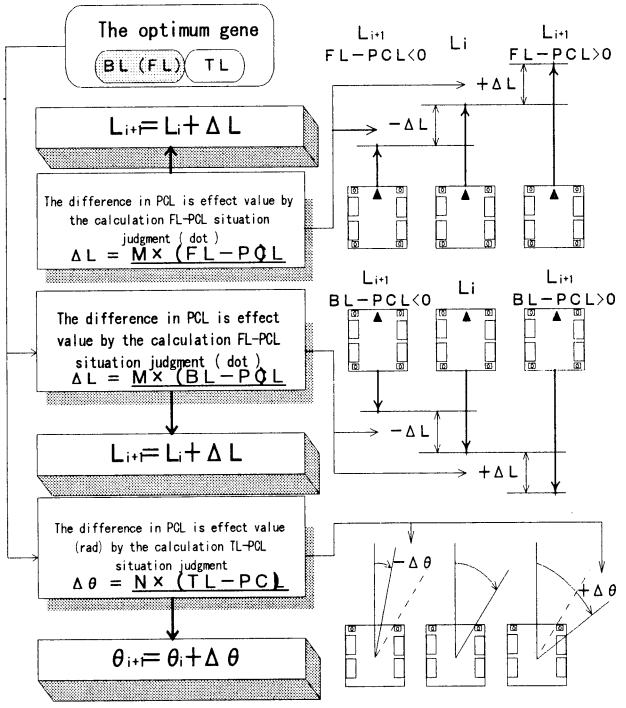


Fig. 3 Flow chart based on genetic algorithm.

In the genetic algorithm processing reflected for controlling value in which it will come by next calculation the highest fitness optimum so. The sum of present turning angle θ i and effect value N by the situation judgment and $\Delta\theta$ which multiplied TL-PCL. By the similar method L_{i+1} of advance and regression distance request and actual control value of the independence travel survey robot the latency of advance and regression distance it handles handle. as turning angle θ_{i+1} after it reflects the gene this turning angle actual control value of the micro mouse the latency of the turning angle it handles handle. Effect value N,M used by this calculation fluctuates by the situation judgment rule pattern.

4 Results and discussions

4.1 Experimental result

In the study, an inspecting experiment depending on the developed robot is carried out. Here, the CCD camera on the robot is used to inspect image on welding surface. The obtained pictures are sent to a host computer by ATM network. And also, these picture can be turn to a web server computer, so that inspecting users can observe the damage on the welding surface by using a client computer. On the other hand, the image recognition of the inspected picture also be carried out by a general software Dr.View[8]. Figure 4 shows the inspected image on the welding surface based on image recognition.

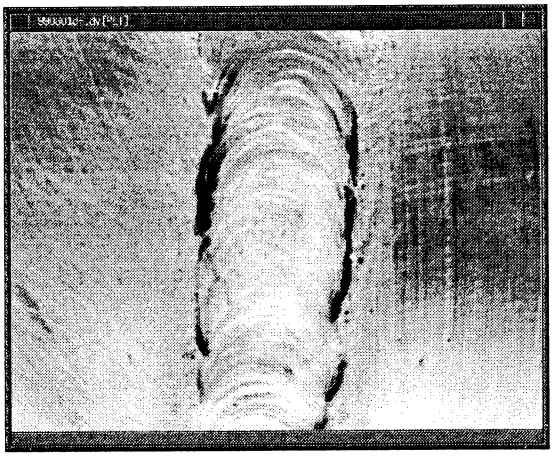


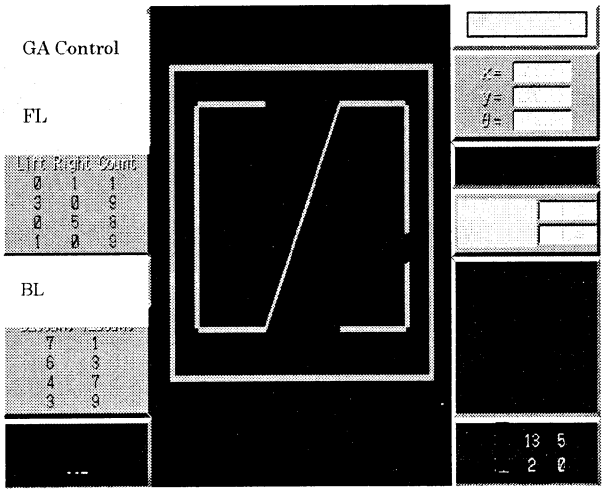
Fig. 4 Inspected image on the welding surface.

4.2 Results of simulation

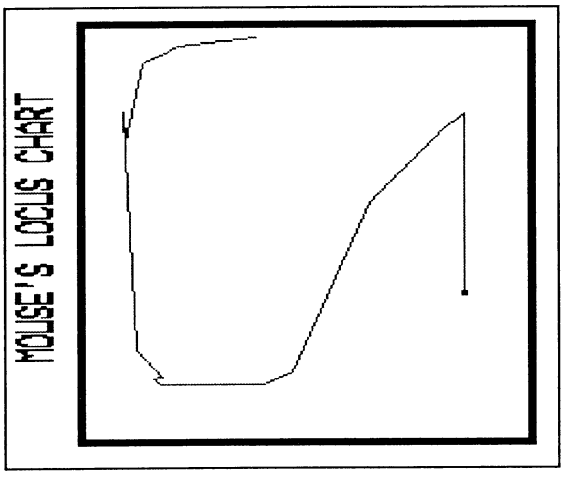
In order to verify effectiveness of the GA algorithm, in the simulation of the navigation, the inspecting robot is made to draw picture, and the color of the course coordinate position of both sensors of obstacle and light is made to distinguish it course obstacle. The coordinate which advances at as it is angle 1 dot in advance is moved, when coordinate position of the sensor of the autonomous travel survey robot is different from course or color of the obstacle. The calculation for making the GA processing is advanced and is declined and is whirled and is moved, and the quantity required by the calculation by doing the GA processing, is advanced and is declined and is whirled and is moved, when it was a

color of the obstacle.

The situation judgment factor is renewed, when non-perception of the sensor followed the case in which the uniformly space and loop was done. The central coordinate of the robot in detecting by obstacle and both light sensors is save as a travel data of the untreated. The parameter is turning angle simulation program the initial stage 1 dot (7.5mm) the initial stage advance regression distance in the initial stage. It is shown at Fig.5(a) and (b) as result of carrying out the simulation from these and the result that it limited the count number in order to require sake and optimum situation judgment factor which deviate from the course, when the navigation of the mobile robot runs at to some extent angle on the course, and carried out the simulation.



(a) Screen presentation in simulation



(b) Result of simulation

Fig.5 Performance of Simulation

5. CONCLUSIONS

In this paper, we develop a autonomous robot system for inspecting damage in welding joint in which imitates intelligent learning. As control tasks of robot become ever more demanding, more intelligent controller are needed. As the optimizer method, the genetic algorithm is offered to utilize optimization of control parameters in navigation of the robot. Our research work developed a new technique for GA involving with a local learning method. Based on some successful performances of simulation and experiment, we know the proposed method can be applied to inspection of welding joint and autonomous navigation in the inspection.

Acknowledgments :

The support provided by the high technological research center of Saitama Institute of Technology in carrying out this work is gratefully acknowledged.

Reference

[1] D. Floreano and F. Mondada, Evolution of homing and navigation in a real robot, Transactions on Systems, Man and Cybernetics, Vol.26, No.3, pp.396-407(1996)

[2] Rodney A. Brooks, "A Robust Layered Control System for a Mobile Robot", IEEE Journal of Robotics and Automation, Vol. RA-2, No. 1, pp.14-23, 1986.

[3] S. Mahadevan and J. Connell, "Automatic Programming of Behavior-based Robots using Reinforcement Learning", Aritifial Intelligence, Vol. 55, pp.311-365, 1991.

[4] D.Y. Ju, K. Toyota, M. Nemoto and T. Iwata, Integration of Fazzy System and Genetic Algorithm and Its Application to the Autonoumous Navigation of Mobile Robot, *Journal of Bejing Mathematics*, Vol.4, No.2, pp.16-25(1998)

[5] A. Varsek, T. Urbancic and B. Filipic, "Genetic Algorithms in Controller Design and Tuning", IEEE Transaction on Systems, Man, and Cybernetics, Vol. 23,No.5, pp.1330-1339, 1993.

[6] D.Montana, "Strongly Typed Genetic Programming", Evolutionary Computation, Vol. 3, No. 2, pp. 199-230, 1995.

[7] John H. Holland, Genetic Algorithms, Scientific American, 1992.

[8] Users Manual of Dr.View, Version 5.0, AJS. Ltd., (1997)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Real-Time Image Processing and Lane Recognition algorithm for autonomous Driving

S. T. Park*, S. Y. Yang*, and J. H. Jung*, J. I. Bae**, M. H. Lee***
* School of Mechanical and Automotive Engineering, University of Ulsan
** Department of Electrical Engineering Pukyong National University
*** Department of Mechanical Engineering Pusan National University

School of Mechanical and Automotive Eng
University of Ulsan
P.P.Box18, Ulsan 680-749, Korea
e-mail soonyy@uou.ulsan.ac.kr

keyword: Lane Recognition, Inverse Perspective Remapping, Autonomous vehicle

Abstract

In this paper, an algorithm to recognize driving lane from road image is presented, which is used for an autonomous driving vehicle. The algorithm is applicable to lateral control of the vehicle in real-time. The first-stage of the process includes grey level digitization, the edge detection of white lane markers and inverse perspective remapping. The first-stage of followed by lane detection process which decides parameters of quadratic polynomial assumed as a road model. These parameters of the road model are determined by Maximum Likelyhood parameter Estimation Method, which has advantages in estimating variables of maximum probability. Also, the estimation method is implemented by Metropolis Algorithm on computer system.

The information of driving lane can be obtained to the driving system every 0.035 seconds. Also, it has been shown to robustly recognize driving lane even in the presence of shadow, discontinuity of lane marker and varied lighting condition.

1. Introduction

Research of automated automobile driving made process to improve a vehicle's safety and convenience by reducing driver strain. The Key Technology of them is the recognition of road surroundings.

To recognize the vehicle's surroundings, we need a sensing system instead of human's organization such as human's eyes. The image processing system with a vision sensor is focused by reason of not requiring additional setting, high accuracy in a near distance, and large measurement range. In this paper, we present the image processing algorithm of lane recognition that is needed for the infrastructure as well as autonomous driving.

Because that raw image from vision sensor involved many information such that lanes, obstacles, other vehicle, and the other objects that can be found in a road surrounding. The image processing algorithm to enhance white line and is used to be recognized white line is required to get a driving course of autonomous vehicle.

2. Image processing

The object of recognition is left lane and right lane on flat plane of ground.

Before explanation of image processing, road is constrained like highway which separate region by white line.

Image processing can be separated as preprocessing, recognition processing.

In preprocessing, white lanes of image which is acquired with vision sensor is transformed and enhanced by Inverse Perspective processing and edge-processing.

It is give some convenience of the recognition processing of lane.

Then, in the recognition processing, we uses a deformable template approach. A parametric family of shapes describes the set if all possible ways that the lane edges could appear an image. A function is defined whose value is proportional to how well a particular set of lane shape parameters matches the pixel data in a specified image. Lane detection is

performed by finding the lane shape parameters that maximize the function for the current image.

2.1 Preprocessing

Raw image from vision sensor may include may other information except road lane. So in order to provide convenience of lane recognition, we may require a operation for enhancing white lane by means of utilizing lane's property before recognition processing.

It's processing is consisted of Inverse Perspective processing and edge processing by constant width property

2.1.1 Inverse Perspective processing

The raw image form vision sensor contain perspective effect by projecting 3D world's objects into vision sensor of 2D plan. To cope with this problem a geometrical transform has been introduced; it allows to remove the perspective effect form the acquired image, remapping it into a new 2-dimensional domain in which the information content is homogeneously distributed among all pixels, therefore, by means of inverse perspective remapping into new 2D plan, the lane information of raw image can be converted into simpler patterns of original geometric shape. Furthermore, road lanes are represented by a constant width value. We can easily utilizes a constant width property of lane's properties and coordinate transform.

As supposing that road is flat, the equation of inverse perspective transform is represented by eq(1) that define by simple 3×3 matrix equation.

X, Y is remapping dimensional coordinate, and x, y is raw image coordinate from vision sensor.

a_1, a_2, a_3, b_1, b_2, b_3, and c_1, c_2, c_3 is defined by the specific acquisition conditions(camera position, orientation, optics, ...) or four more (x, y)s and matched (X, Y)s which were known previously.

$$\begin{pmatrix} \lambda X \\ \lambda Y \\ \lambda \end{pmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad ---(1)$$

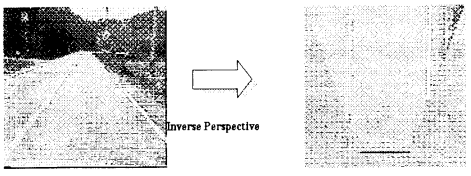


Fig. 1 raw image and remapping image.

2.1.2 Edge-processing with a constant width

Road lanes have higher gray level than other road surfaces and edge that is changed quickly from low gray levels to high gray levels. Furthermore, it's pattern is dark-bright-dark gray level's style with a constant width. It means that a value that is differentiated in horizontal direction have a constant distance between a positive peak value and negative peak value. In additional, a positive peak value means dark-bright pattern, and a negative peak value means bright-dark pattern too.

In generally, central differential is used for horizontal edge-processing.

It express that

$$\Delta f(x_i) = \frac{(f(x_{i+1}) - f(x_{i-1}))}{\Delta x} \quad ---(2)$$

It means that a differential value at point x_i is the value which was subtracted point x_{i-1} from point x_{i+1}.

So, if a positive peak value may expand to right sides, and a negative peak value may also expand to left sides, and be added to positive sides by absolute values, then we can obtain better results than typical edge processing's results.

Edge-processing with a constant-width at the center of image's line have a 8 pixel-width is expressed by next equation.

$$\begin{aligned} F(x_i) = {} & +\Delta f(x_{i-3}) + \Delta f(x_{i-2}) + \Delta f(x_{i-1}) \\ & - \Delta f(x_{i-1}) - \Delta f(x_{i-2}) - \Delta f(x_{i-3}) \\ F(x_i) = {} & -f(x_{i-4}) - f(x_{i-3}) + f(x_{i-1}) + \\ & 2f(x_i) + f(x_{i+1}) - f(x_{i+3}) - f(x_{i+4}) \end{aligned} \quad ---(3)$$

This equation can be expressed by matrix [-1 -1 0 1 2 1 0 -1 -1] which is called 'mask' in image processing.

If the width of line is 6 pixel, then we can easily expect that [-1 -1 0 1 2 1 0 -1 -1] is changed into [-1 -1 1 2 1 -1 -1].

This processing have the advantage of enhancing lines of a horizontal constant-width

It's processing results that is adopted at the image that is transformed by inverse perspective remapping is showed at fig 2.

And it's convolution-job can be realized on efficient implementation by SIMD machine such as Intel's

Pentium MMX CPU.

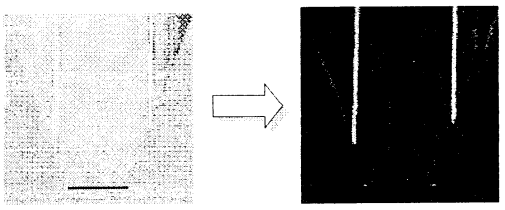


Fig 2. The image of edge-processing with a constant width

2.2 Recognition processing

Recognition processing approach with three components.

-A parametric family of shapes which describes all possible ways that object can appear in the image.

-A likelihood function which measures how well a particular hypothesised object shape matches a given image

-A method for finding the shape parameters that maximizes the likelihood function for the image being examined

2.2.1 Prior Model of Lane Shapes

Assume lane's curvatures is small and continuous, lane's feature can be represented by parabolic lines.

$$x=0.5ky^2+ay+b \qquad ---(4)$$

By Inverse perspective remapping, we can obtain a image which is removed perspective effect. The shape of lane in this image is corresponded with the shape of lane on ground plane.

For left and right lane edges is made that the arcs have equal curvature and equal tangential orientation where they intersect the X axis, so k and a is will be equal for the left and right lane. As a result, the lane shape in an image can be defined by the four parameters k, b_{left}, b_{right}, and a

2.2.2 The likelihood function

The intuition underlying the likelihood function is that there should be a brightness gradient near every point along the lane edges. The larger the magnitude of that gradient. The more likely it is to correspond to a lane edge. Also a distance between left lane and right lane is distributed from minimum 2.75 to maximum 3.75. Then The likelihood function is defined as eq(5)

$$\left[\sum_{i,j}\left\{GradMag_{pixel}\times\frac{1}{1+a*DisL_{pixel}}\right\}+\sum_{i,j}\left\{GradMag_{pixel}\times\frac{1}{1+a*DisR_{pixel}}\right\}\right]\times[atan(b_{right}-b_{left}-2.75)-atan(b_{right}-b_{left}-3.75)] \qquad ---(5)$$

Where, $DisL_{pixel}$ is the distance in columns from the closest left lane edge, $DisR_{pixel}$ is it of right lane edge and a determines how fast $\frac{1}{1+a*x}$ decreases as x increase.

2.2.3 Metropolis Algorithm

In general, the function of eq(5) is no a concave function, and often contains several local maximum. As a result, local optimization technique such as gradient descent are no appropriate. The current implementation uses the Metropolis algorithm with a geometric annealing schedule to perform this maximization.

All the results shown in this paper were generated by running the Metropolis algorithm for 80 iterations. The initial temperature is 100, and the final temperature is 1.65.

3. Experiment and results

This image processing is tested in experiment system consists of Pentium II 266㎒ and Meteor Image Grab board which produced for low-cost Model in Matrox corporation.

This Metro model can manage vision information of 640×480 resolution at 30 frame/sec. For satisfying of image grab board speed, Vision control step must be within 30ms. We optimized image processing algorithm by implementing MMX assembly instruction code with reason of having a difficult to speed up by using general C code. The Image processing time is reduced to 13- 20 times than before optimization.

For testing in various road conditions, we record the image of road in video tape by driving the car that attached camcoder. Then image processing is tested by playing video tape in the lab.

In this case, we cans successfully obtain a road lane's informations except that road lanes went outside of view region in a large curvature and a view contained a big disturbance like crossroad , the

mark of road's surface, and etc.

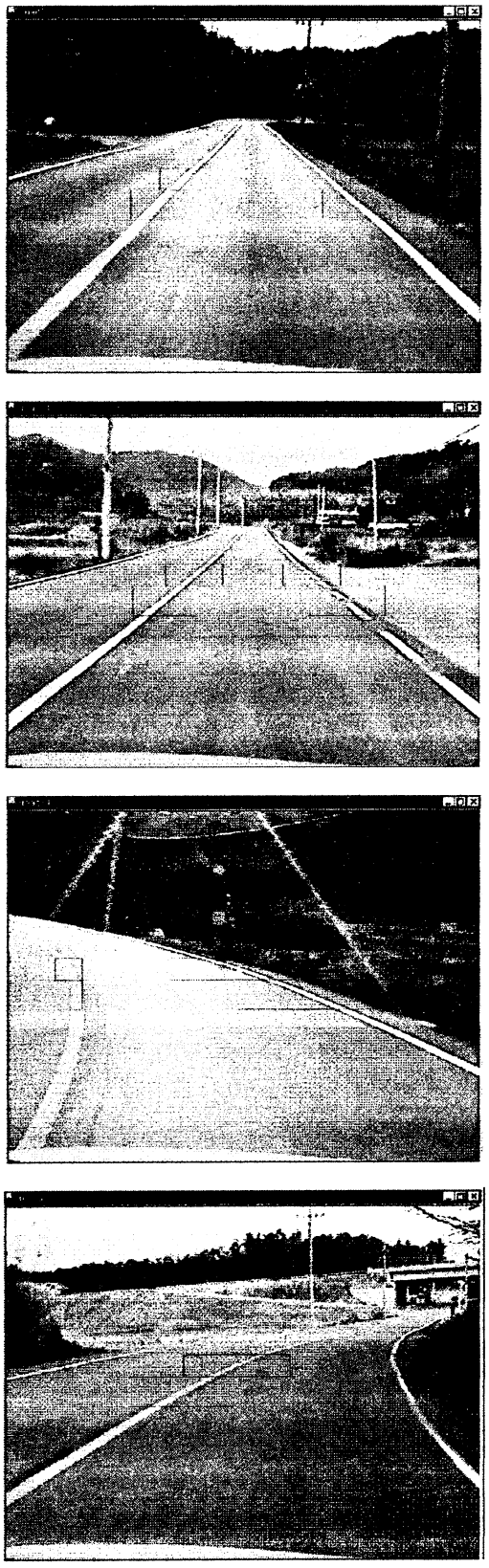


Fig 3. Results of recognition of lanes

4. Conclusion

This lane recognition system can successively and reliablely extract a lane's information by real-time in various environment such as the presence of shadow, discontinuity of lane marker and varied lighting condition. And all processing is finished for 30 ms.

This system will be available with lane departure warning system and lateral control for autonomous vehicle.

Acknowledgment

This research was supported by Research Center for Machine Parts and Materials Processing(REMM) at University of Ulsan.

References

[1] Chris Kreucher, Sridhar Lakshmanan, and Karl Kluge, "A Driver Warnig System Based on the LOIS Lane Detection Algortithm" IEEE International Conference on Intelligent Vehicles, 1998

[2] 日本機械學會, "RC141 ITS-AVCSに 關する 研究報告書", JSME, 1998.5.31.

[3] 景山, 渡辺, "自動操縦車用畵像處理技術に 關する基礎的研究",日本機械學會論文集C編, 1996 Vol.62, No.593

[4] M. Bertozzi and A. Broggi, "GOLD: a Parallel Real-Time Stereo Vision System For Generic Obstacle and Lane Detection" IEEE Transactions on Image Processing 7, January 1998, pp. 62-81

[5] H. A. Mallot, H. H. Bülthoff, J. J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical flow computation and obstacle detection" Biological Cybernetics 64, 1991, pp. 177-185

[6] M. Chen, T. Jochem, and D. Pomerleau, "AURORA: A Vision-Based Roadway Departure System, in Proceedings, IEEE Conference on Intelligent Robots and Systems, vol. 1, August 1995, pp. 243-248

[7] Intel Corporation, Intel Architecture MMX Technology Developer's Manual, Intel Corporation, 1997. avalaible at http://www.intel.com

[8] J. Serra, "Image Analysis and Mathematical Morphology", Academic Press, London, 1982.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

The development of inverter for generation of electricity from solar panel

Masanori Sugisaka and Koji Sakamoto
Department of Electric and Electronic Engineer
Oita university
700 Dannoharu, Oita 870-1192,JAPAN
Tel:+81-975-54-7831 Fax:+81-975-54-7820

E-mail:msugi@cc.oita-u.ac.jp

Abstract

In the 20th century, our industry and living standard were developed remarkably based on fossil fuel such as petroleum and coal. However, there is not much fossil fuel. When dusts burn they have caused various pollution problems. The avoidance from fossil fuel is future theme because electric power is more consumed in recent years.

In this paper, the efficiency and the mechanism of the operation are investigated using a small inverter that we manufactured. The research results are applied for the generation of electricity.

1. Introduction

Various ways of the generation of electricity are accomplished using fossil fuel. The generation of electricity from solar panel is remarkable as one of the clean energy. At present stage, the efficiency of transformation from solar energy to electricity is not good because it is influenced by both the weather and material. Also there is a problem such that a cost to produce the electricity is too high, and so on. From a point of view of protecting environment, it is expected when the development of the research on the generation of electricity from solar panel is expected.

The system of the generation of electricity from solar panel is mainly composed of solar panel, battery, inverter and controller. Inverter is playing the important role which changes direct current voltage into the alternating current voltage.

2. The composition of the circuit

The source transmission which decides frequency of the power supply needs twice of frequency(120Hz) and uses NE555 as the multivibrator.

In Fig.1, a part of the resistance R_B is replaced by variable resistance to change frequency of oscillator (from 90Hz to 130Hz) a little leeway given by

$$f = 1.44/[C/R_A + 2R_B]$$

Solution of R_B is represented by

$$R_B = (0.72/Cf) - R_A/2$$

If R_A is 10kΩ and C is 0.1μF, it need

to change R_B (from 50.5k Ω to 75.2k Ω).

Fig.2 is the oscillator circuit to drive in the gate of power MOSFET.
A range of frequency of oscillator of NE555 is obtained from 72.2 to 145 Hz in this figure.

IC_2 is called dial D flip flop. By returned output $\overline{Q}$ to input D, it is the circuit which exactly makes input frequency correctly 1/2, the duty ratio of output wave transmits is 50 % correctly.

IC_3 is the circuit of dead time occurrence circuit not to move MOSFET the same time.

It is composed of output $Q, \overline{Q}$ in IC_3, delay circuit output with 10 k Ω of resistance, and condenser 1000pF in the AND circuit.

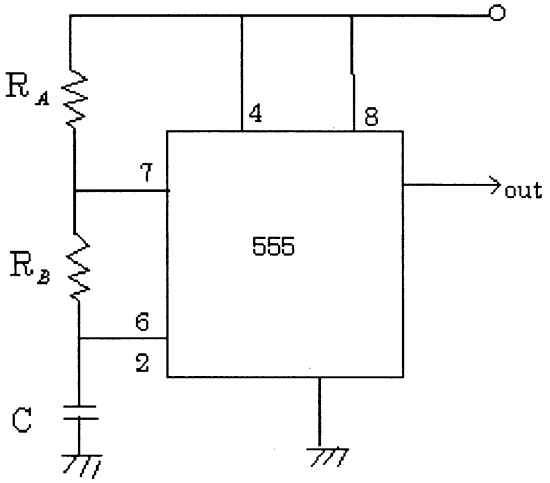


Fig. 1 Multivibrator

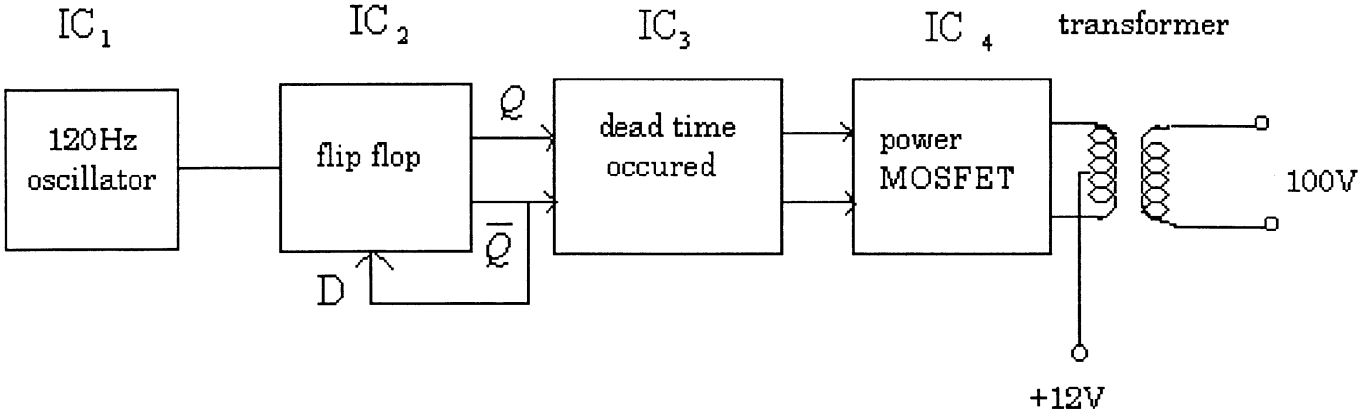


Fig.2 Oscillator circuit

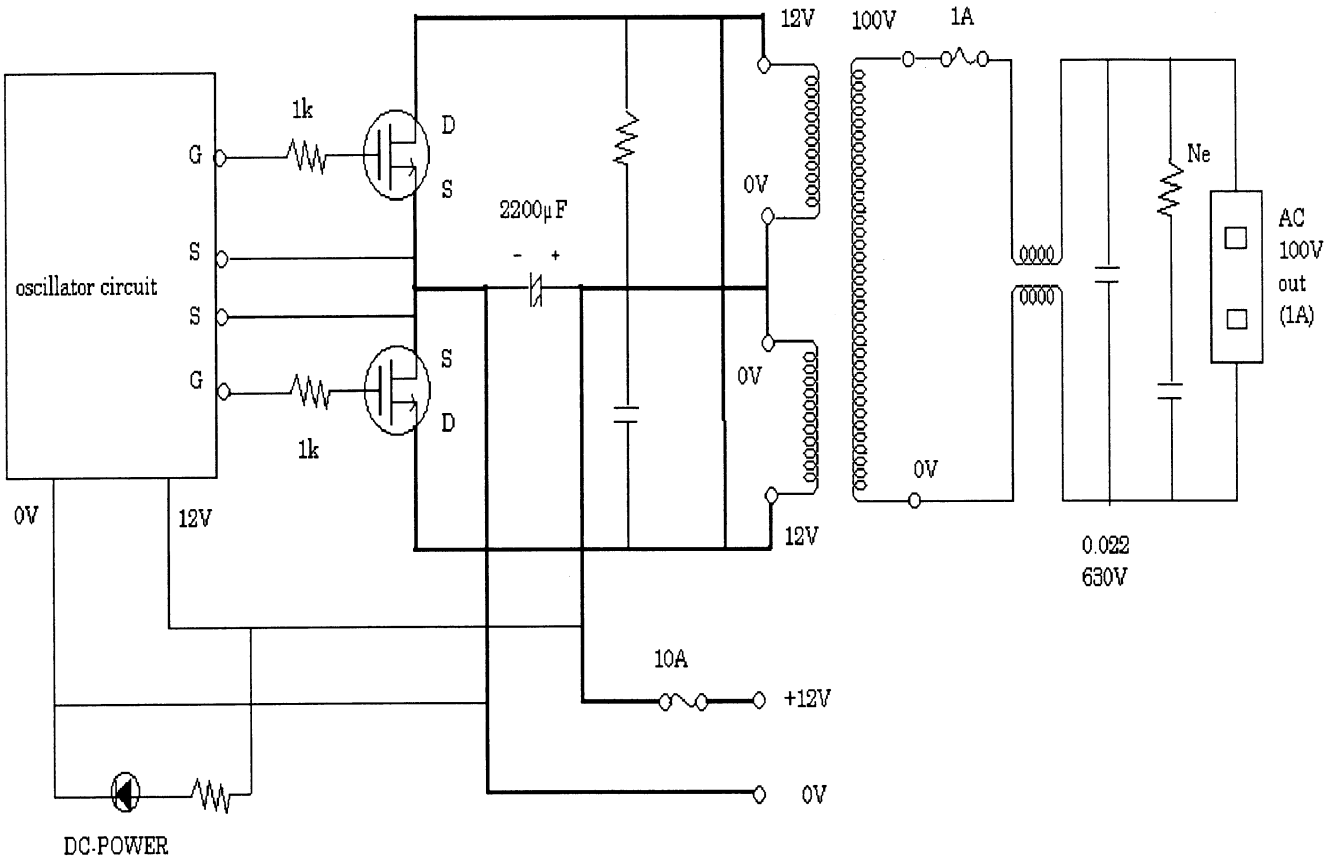


Fig.3 Inverter circuit

In Fig.3, a whole circuit is shown. The mechanism of this circuit is a push pull amplification circuit with good electric power efficiency.

This circuit makes an alternating current waveform occur by inputting two different square waveforms alternately.

As the input power supply, one of DC 12 V battery for the automatic car is used instead of the accumulator for the solar generation of electricity.

It produces the stable voltage 9 V with DC 12 V using three terminal regulator. This voltage is supplied to parts of IC of oscillator circuit in order to make a gate signal. The current is amplified to produce square wave of

alternating current by power MOSFET to produce DC 100 V using the transformer.

To prevent from getting into high frequency, parallel condenser and series coil are inserted into filter.
The reason for using power MOSFET is that the price is cheep and there are few electric power losses at the gate.

3. Series resonance filter

We want to make a sine wave because we are thinking of selling electricity in future. But it is possible to use the square wave sufficiently. The one way to produce a sine wave is to use series resonance filter.

If the switching frequency ω_s is equal to the resonance frequency ω_o, in the case that of filter Q is high, the getting into ingredient of high frequency wave is removed and becomes a sine wave is obtained approximately.

4. Conclusions

An electric light was stuck if trying in the fluorescence light for the desk.

The distortion was terrible in the part of switching when we saw an output corrugation.
It may not be influent like the fluorescence light, but it is thought that noise is influent on the radio and the television .

Also, we found that it is difficult to make a sine waveform as the degree of power supply. However, it is possible to use this.

Future problem is out of noise and making the bigger electric power inverter.

References

[1] Kosuke Harada, Tamotu Ninomiya, Tadao Nakano: "Fundamentals of Electronic Circuit" Corona Publishing Co.,Ltd,pp.35-72

[2] John G. Kassakian, Martin E. Schleche, George C. Verghese: "Principle of Power Electronics "Addition-Wesley Publishing Company, Inc. 1997,pp.15-42 and pp. 131-148

[3] H. Omori, M. Nakaoka, H. Yamashita, T. Maruhashi: "A Novel Type Introduction-Heating Single-Ended Resonant Inverter Using New Bipolar Darlington Transistor" in IEEE Power Electronics conference(1985) ,pp-590-599

[4] K-H.Lie, R. Oruganti, F. C. Lee: "Resonant Switches-Topologies and Characteristics" in IEEE Power Electronics Conference (1985), pp.106-116

[5] S. K. Ghandi: "Semiconductor Power Devices; Physics of Operation and Fabrication Technology New York: John Wiley & Sons (1977),pp.124-134

[6] D. M. Divan: "Design Considerations for Very High Frequency Resonant Mode dc/ac Converters" IEEE Trans. Power Electronics(1987)pp.45-54

[7] R. J. King and T.A. Stuart: "Modeling the Full Bridge Series Resonant Power Converter" IEEE Tran. Aerospace and Electronic Systems,pp.449-460

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Switching Petri Nets as a tool for User Interface

Yosuke Funami
Dept. of Computer Controlled Mechanics
Shonai College of Industry and Technology
3-57-4 Kyoden, Sakata 998-0102 Japan

Kou Yamada
Dept. of Elec. and Inf. Eng.
Yamagata University
4-3-16 Jonan, Yonezawa 992-8510 Japan

Abstract

This paper examines a problem to expand Petri Nets that are a useful tool as User Interface. The proposed expansion of traditional Petri Nets is called Switching Petri Nets. In Switching Petri Nets, event of systems corresponding to Transitions depend on the color of Colored Tokens. Switching Petri Nets can represent several systems. The usefulness and some characteristics of Switching Petri Nets are shown.

1 Introduction

This paper examines a problem to expand Petri Nets that are a useful tool as User Interface. Petri Nets were proposed by C.A.Petri in 1962 at first time [1]. Petri Nets are a mathematical model or tool for representing discrete event systems such as parallel systems, asynchronous systems, nondeterministic systems or their combined systems. Petri Nets can represent discrete event systems graphically and simply. In addition, discrete event systems that is represented by Petri Nets can be analyzed and evaluated of their characteristics. Therefore Petri Nets are applied for many fields of applications, for example, application for scheduling or modeling of manufacturing systems [2, 3, 4, 5, 6], modeling of computer job flow [7, 8, 9, 10], verification of specifications of communication software [11] and so on. All of these applications are using the property of characteristics of Petri Nets such as:

- Petri Nets can describe systems simply and graphically, that is, Petri Nets can be used as a graphical language.
- A system described using Petri Nets can be analyzed.
- Petri Nets tools have developed.

In this way, Petri Nets is a useful tool as describing several discrete event systems. When some discrete event systems is described using Petri Nets, in some case, the system can not described well. For such systems, extension of Petri Nets have been considered. Some of them are Colored Petri Nets, Hierarchical Petri Nets, Timed Petri Nets and so on. Petri Nets to describe discrete event systems has been developed.

Though Colored Petri Nets which are useful for complicated systems, they have a delicate matter. This paper presents examples that can not be represented using traditional Petri Nets. In addition, to represent such systems, we propose new expansion of traditional Petri Nets, that is so called Switching Petri Nets. In Switching Petri Nets, event of a system corresponding to Transitions depend on the color of Colored Tokens. That is, when the firing of one Transition, if the color of Colored Tokens is blue then the system operates event A, and if the color of Colored Tokens is red then the system operates event B. Besides a concept of Hierarchy is introduced into Switching Petri Nets. Such as extended Petri Nets are called Hierarchical Switching Petri Nets. The fires of Hierarchical Switching Petri Nets imply execution of event. Transitions of Hierarchical Switching Petri Nets have information on Partial Nets. When Transitions of Hierarchical Switching Petri Nets fire, these can choose which Partial Nets execute by the color of Colored Tokens. That is, in the fire of one Transition, if the color of Colored Tokens is blue then Hierarchical Switching Petri Nets execute Partial Nets A, and if the color of Colored Tokens is red then Hierarchical Switching Petri Nets execute Partial Nets B. Hierarchical Switching Petri Nets can represent more simply than Switching Petri Nets.

2 Application of Petri Nets to Robot Control

In this section, a relation between elements of Petri Nets and jobs of Robot when Petri Nets is applied to Robot control is described. In addition, the objectives in the present paper are clarified.

When Petri Nets are applied to operate real systems, the fire of Transition implies Robot practices something to job. The job Robot practices had already decided in each Transitions. In a case that Robot Arm is operated, the relation between jobs of Robot Arm and the fires of Transitions is described next. There is a part in front of Robot Arm. The operations is Robot Arm goes to the part and grips it. The operations described by Petri Nets is showed in Fig. 1 . But, the fires of Transitions in Fig. 1 correspond to

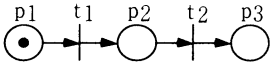


Figure 1: The operations described by Petri Nets

jobs in Tab. 1 .

Table 1: Transitions and the jobs

Transitions	Jobs
t_1	goes to the part
t_2	grips the part

From Fig. 1 and Tab. 1 , each Transitions have only one predetermined job.

Assuming that there are several sections in front of Robot Arm. The number of sections is denoted by n. Operations are to let Robot Arm to move one section and grips it. The operations described by Petri Nets is shown in Fig. 2 . Petri Nets shown in Fig. 2 have Conflict. Here, the fires of Transitions t_{1-1},$t_{1-2}\cdots t_{1-n}$ are equal to the jobs in Tab. 2 .

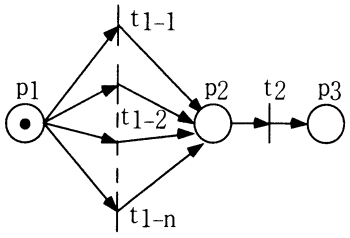


Figure 2: Petri Nets having Conflict.

However, Petri Nets having Conflict do not know which Transition fires. In this case, for example, we must decides which Transition fires. Since the more the number of sections increase the bigger Petri Nets

Table 2: Transitions and the jobs

Transitions	Jobs
t_{1-1}	goes to No.1
t_{1-2}	goes to No.2
$\cdots$	$\cdots$
t_{1-n}	goes to No.n
t_2	grips the part

is, Petri Nets can not represent such operations simply. To overcome this problem, we propose switching Petri Nets.

3 Switching Petri Nets

In this section, we propose switching Petri Nets. Swiching Petri Nets expand Transitions of the traditional Petri Nets. We call Transitions in Switching Petri Nets Switching Transitions, and denote simply st. This concept is effective to operate several jobs in only one Petri Nets.

3.1 Switching Transitions

Switching Transitions which depend on Colored Tokens defined in this subsection. Switching Transitions have some jobs corresponded to the color of Colored Tokens.

Definition 1 *Switching Transition st is a function. It's finite set is*

$$ST = \{\ st_1, st_2, \ldots, st_n\ \}.$$

A finite set of Colored Tokens is

$$TK = \{\ tk_1,\ tk_2,\ \cdots,\ tk_m\ \}.$$

Then, A finite set of jobs which one Switching Transition st_i has already depends on Colored Tokens,

$$v_i = \{st_i(tk_1), st_i(tk_2), \cdots, st_i(tk_m)\ \}.$$

And, a finite union of the finite set of jobs which each Switching Transitions have is

$$V = \{\ v_1, v_2,\ \cdots,\ v_m\ \}.$$

Here, Switching Transitions are applied to the system, there are several parts in front of Robot Arm, described hereinbefore. If Robot Arm goes to the part of No.i and grips it, the color of Colored Token is tk_i. Then, Switching Petri Nets described in Fig. 3 . Here,

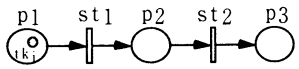


Figure 3: Switching Petri Nets

Table 3: The fires of Switching Transitions and the jobs

	st_1	st_2
$st(tk_1)$	goes to the No.1	grips the No.1
$st(tk_2)$	goes to the No.2	grips the No.2
...	...	...
$st(tk_n)$	goes to the No.n	grips the No.n

the jobs of Robot Arm corresponding to the fires of Switching Transitions are showed in Tab. 3 . The job corresponding to the fire of Switching Transition st_2 do not depends on any Colored Tokens, this job is that Robot Arm grips the part.

By letting Transitions have several jobs corresponding to Colored Tokens as Tab. 3 , the operations of Robot Arm can be described very simply as Fig. 3 .

3.2 Definitions of Switching Petri Nets

This subsection gives the definition of Switching Petri Nets.

Switching Petri Nets are Nets defined next.

Definition 2 *A structure SN of Switching Petri Nets is a term of 8 members, SN(P,ST,V,TK,I,O,C,W). P is a finite set of Places. P and ST are disjoint, that is* $P \cap T = \emptyset$. *Assuming that a set of input Places of Switching Transitions is* P_{st} *and that a set of output Places of Switching Transitions is* ${}_{st}P$, $I{:}ST \rightarrow P_{st}$ *is Input Function and* $O{:}ST \rightarrow {}_{st}P$ *is Output Function.* $C{:}P \rightarrow \mathcal{N}^+ \cup \infty$ *is Capacity Function. Here,* $\mathcal{N}^+$ *is a set of positive numbers. Using binary relation* $F \subseteq (P \times ST) \cup (ST \times P)$ *showing flows in the Nets,* $W{:}F \rightarrow \mathcal{N}^+$ *is Weight Function.*

3.3 Marking and Fire rules of Switching Petri Nets

This subsection says Marking and Fire rules of Switching Petri Nets.

Assuming that the number of all Colored Tokens in Place p_i is m_i, Marking M is $M(p_i)=m_i$. And $M(p_i,tk_j)$ gives the number of Colored Tokens tk_j in Place p_i.

In Marking M, if Switching Transitions are enable, Switching Transitions fires. Fire rules of Switching Transitions is the next definition.

Definition 3 *For all of Input Places* $p_i \in P$ *of Switching Transition* $st_j \in T$ *and for all of Output Places* $p_k \in P$ *of Switching Transition* st_j, *Switching Transitions* $st_j \in ST$ *in Switching Petri Nets SN(P,ST,V,TK,I,O,C,W) having Marking M is enable, only if*

$$M(p_i,tk_h) \geq W(p_i,I(t_j))$$

and

$$C(p_k) \geq W(p_k,O(st_j)) + M(p_k,tk_h).$$

If Switching Transitions which is enable have fired, Colored Tokens move on Places in Nets. Then, Marking M becomes new Marking M'.

Definition 4 *Switching Transition* st_j *in Switching Petri Nets having Marking M can fire anytime, if it is enable. When Switching Transition* st_j *have fired, Marking*

$$M'(p_i) = M(p_i) - W(p_i,I(t_j)) + W(p_i,O(t_j))$$

and

$$M'(p_i,tk_h) = M(p_i,tk_h) - W(p_i,I(t_j)) + W(p_i,O(t_j))$$

is given.

4 Hierarchical Switching Petri Nets

In this section, Switching Petri Nets are extended. Extended Switching Petri Nets have a combination of Switching Petri Nets and Hierarchical Petri Nets. We call such Petri Nets Hierarchical Switching Petri Nets.

A finite set of Partial Nets is assumed to be denoted by

$$H = \{ h_1, h_2, \ldots, h_n \}.$$

And jobs which Switching Transition have corresponds to Partial Nets. That is, the fire of Switching Transition on Colored Tokens is equal to execution of Partial Nets. A relation between Colored Token tk_i and Partial Nets h_i is written by

$$st(tk_i) = h_i.$$

Next a example of Hierarchical Switching Petri Nets is shown. In the case that one Robot manufactures section A or section B, without loss of generality two jobs are assumed to be different each other. This type of

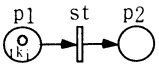


Figure 4: Hierarchical Switching Petri Nets

system can be represented using Hierarchical Switching Petri Nets simplly. A example is shown in Fig. 4 . Here, Partial Nets to manufacture the part A are h_1, Partial Nets to manufacture the part B are h_2. Put Colored Token tk_1 on Place p_1 when Robot manufactures the part A, and put Colored Token tk_2 on Place p_1 when Robot manufactures the part B. As Fig. 5 , the fire of Switching Transition st depends on the color of Colored Tokens, and executes Partial Nets h_1 or h_2.

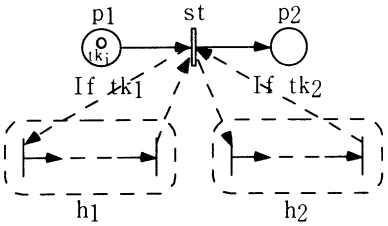


Figure 5: Execution of Partial Nets

5 conclusion

In this paper, we proposed switching Petri Nets and Hierarchical Switching Petri Nets that is expansion of traditional Petri Nets. The usefulness and some characteristics of switching Petri Nets and Hierarchical Switching Petri Nets were also shown.

References

[1] C. Petri, "Fundamentals of a theory of asynchronous information flow", *Information Processing '62, Proc. Of the 1962 IFIP Congress*, 386/390,1962.

[2] N. Komoda et al., "An autonomous decentralized control system for factory automation", *IEEE Computer*, 17-12, 73/83,1984.

[3] R. Vallete et al., "Putting Petri Nets to work for controlling flexible manufacturing systems", *Proc. ISCAS 85*, 929/932,1985.

[4] T. Murata et al., "A Petri Net based controller for flexible and maintainable sequence control and its applications in factory automation", *IEEE Trans. on Industrial Electronics*, IE33-1,1/8,1986.

[5] K. Hasegawa, K.Takahashi and P.E.Miyagi, "Application of the mark flow graph to represent discrete event production systems and system control", *Trans. of the society of instrument and control engineers*, 24-1, 69/75,1988.

[6] P.E.Miyagi, K. Hasegawa and K. Takahashi, "A programming language for discrete event production systems based on production flow schema and mark flow graph", *Trans. of the society of instrument and control engineers*, 24-2, 183/190,1988.

[7] W.Rising, *Petri Nets in software engineering, in G.Goos and J.Hartmanis(eds), "Petri Nets : Applications and relationships to tther models of concurrency", Lecture notes in computer science 255*, Springer-Verlag, 63/96,1987.

[8] M.K.Molloy, "Performance analysis using stochastic Petri Nets", *IEEE Trans. Computer*, C31-9, 913/917,1982.

[9] N.G.Leverson et al., "Safety analysis using Petri Nets", *IEEE Trans. on Software Engineering*, SE13-3, 386/397,1987.

[10] M.A.Holliday, M.K.Vernon, "Exact performance estimates for multiprocessor memory and bus interference", *IEEE Trans. Computer*, C-36,76/85,1987.

[11] P.M. Merlin, "A methodology for the design and implementation of communication protocols", *IEEE Trans. Communication*, COM24-6, 614/621,1976.

[12] K.Jensen, *Coloured Petri Nets : Basic Concepts, Analysis Methods and Practical Use, Vol.1, EATCS Monographs on Theoretical Computer Science*, Springer-Verlag, 1992.

[13] K.Jensen, *Coloured Petri Nets : Basic Concepts, Analysis Methods and Practical Use, Vol.2, EATCS Monographs on Theoretical Computer Science*, Springer-Verlag, 1995.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Stochastic Diffusion Control for Gene Regulation Protein.

H, Hirayama., N, Kitagawa., *Y, Okita and **T, Kazui
Asahikawa Medical College hirayama@asahikawa-med.ac.jp
*Shizuoka University. **Hamamatsu Medical university.

Abstract

We propose a mathematical method to analyze the transient change in the probability of gene regulation protein particle for the first time arrival at the target operator gene region of the DNA chain.The probability was expressed by the modified Bessel functions. By applying the Allen's approximation for the modified Bessel function, we obtained an approximation form of the first time arrival probability of the gene regulation protein particle immediately after the onset of the reaction. The impulse response of the probability showed considerable oscillation. The present result indicates the genetic expression particularly at its initial phase is unstable.

The present method will be available for evaluating the temporal change in the probability of the gene regulation.

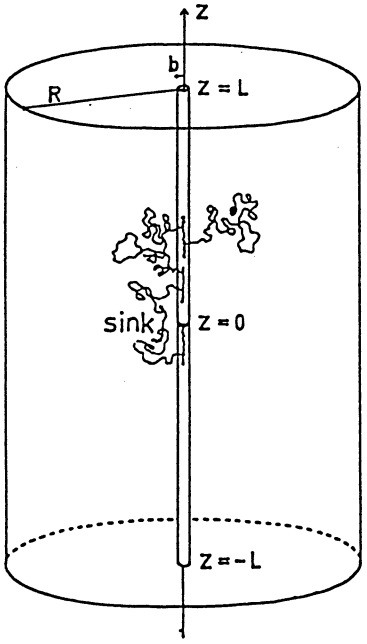
key words : Diffusion control, Gene expression. Chaos.

1. Introduction.

Regulation of gene expression on the DNA is one of the most important issue in the life science. The basic problem, how the gene is regulated at the onset of the gene expression is still in question. Because the lack of rigorous theoretical approach. The present paper introduce recurrent binding model of the gene regulation protein particle, repressor protein for the target gene region on the DNA (the operator gene) which has been proposed by Berg (1976). Unfortunately, however, he did not analyze the transient changes in the probability of binding process of the gene regulation protein.

The present paper introduce a mathematical method for analyzing the temporal changes in the binding process of the gene regulation protein immediately after the onset of the binding reaction.

Fig 1



2. Modeling

2-1. The basic differential equation

We take an initially homogeneous distribution of particles as repressors and sinks as operators in cylindrical coordinates. Every sink is imbedded in the middle of a long cylindrical chain as a DNA molecule of length 2L and radius b. A particle moves with a diffusion constant D in the medium and every time it faces to the chain, the particle is absorbed into the chain and moves along the chain with the a one dimensional diffusion constant D1.

By the stochastic process for short time duration ,dt, with the probability of λ dt, the particle will leave the chain unless it has met the sink in which case it would be permanently absorbed.

The diffusional flow of particles into the chain along, off and again onto the chain and finally into the sink is a complicated coupled multiple dimensional diffusion. For the simplicity, we separate the movement of the particle as

u(z,t) dz : The probability that the particles is in the space interval of (z, z+dz) on the chain at the time t.

G(z,t) dz dt : The probability that a particle for the first time arrivals at the chain in the interval (z, z+dz) in the time interval (t, t+dt)

F(z,z', t-t') dz dt : The probability that a particle leaving the chain from z' at time t' again returns to the chain in the interval (z, z+dz) and in the time interval (t, t+dt)

By the symmetric assumption, analysis was confined only for the half plane $0 < z < L$ of the chain. The diffusion equation along the chain becomes

$$\frac{\partial u}{\partial t} = D1 \frac{\partial^2 u}{\partial z^2} - \lambda u + \lambda \int dz' \int dt' F(z,z',t-t') u(z',t') + G(z,t) \qquad \text{----------(1)}$$

The third term describes the return to the chain of a particle that left it at some earlier time. The last term is the arrival of an uncorrelated particle.

2-2. The boundary and initial conditions.

The boundary condition at z=0 for an ideal sink

$$u(0,t) = 0 \qquad \text{----------(2)}$$

At the ends of the chain the choice of boundary conditions is not crucial and the reflecting boundaries are used

$$\partial u / \partial z = 0 \qquad (z = \pm L) \qquad \text{----------(3)}$$

For no initial particle on the chain, the initial conditions are

$$u(z,0) = 0 \qquad \text{----------(4)}$$

The flux of particles into the sink can be computed by

$$\Phi(t) = D1 \ \partial u / \partial z \qquad (z=0) \qquad \text{----------(5)}$$

2-3. Method of solution.

The solution is given by the diffusion equation for the concentration c(r,z,t) of particles

$$\frac{\partial^2 c}{\partial r^2} + \frac{1}{r}\frac{\partial c}{\partial r} + \frac{\partial^2 c}{\partial z^2} - \frac{1}{D}\frac{\partial c}{\partial t} = 0 \qquad \text{----------(6)}$$

The boundary conditions are

$$\partial c / \partial z \,(z = L, z = 0) = 0 \qquad (7), (8),$$

$$k\,c(r = b) - \partial c / \partial r \,(r = b) = 0 \qquad (9) \text{ and}$$

$$\partial c / \partial r \,(r = R) = 0 \qquad (10)$$

and the initial condition

$$c(r,z,t=0) = \delta(r-a)\,\delta(z-z') / (2a\pi) \text{ ----- } (11)$$

$2\pi Dbk$ would be the association rate onto the chains for a homogeneous distribution of particles.

$$c(r=b) = 0 \qquad \text{----------(12)}$$

The particle flux, F(z,z',t) back to the chain which enters equation (1) is now given by

$$F(z,z',t) = 2\pi bD\,\partial c / \partial r = 2\pi bDkc(r=b) \qquad \text{-------(13)}$$

This can be calculated after solving equation (6) to (12). For the initial conditions, we assume that one particle distributes homogeneously in the cylinder.

$$c(r,z,t=0) = 1 / (\pi(R^2 - b^2)L) \qquad \text{---------(14)}$$

As the solution with (12) is a fundamental solution,

$$G(z,t) = \int_0^L dz' \int_b^R da\, 2\pi a\, F(z,z',t) / [\pi(R^2 - b^2) L] \qquad \text{----------(15)}$$

The understanding of equation (15) is following.

On the homogeneous initial condition (13), the particle start in the distance interval (a, a+da) from the chain by the probability $2\pi a\,da / (\pi(R^2 - b^2))$. With the probability dz' / L, the particle will start in the interval (z', z'+dz'). Integrating the fundamental solution over all z' and a gives the solution G. By using the Laplace tansformation, the solution of equations (6) to (11) is achieved through separating the variables

$$c(r,z,t) = v(z,t)\ w(r,t) \qquad \text{---- -(16)}$$

The axis (z) dependent part is found to

$$v(z,t) = 1/L + 2/L \sum \cos(n\pi z'/L)\cos(n\pi z/L)\ \exp(-Dt(n\pi/L)^2) \qquad \text{-----(17)}$$

The Laplace form of $\Phi(t)$ of particles into the sink is

$$\Phi(s) = L\,G(s) / [\,1 + 2/\{\lambda[1 - \phi(s)] + s\} \sum_{n=1}^{\infty} \{ D1(n\pi/L)^2 + s + \lambda[1 - \phi(s + Dn^2\pi^2/L^2)]\}] \qquad \text{-----(18)}$$

where

$$\psi(s) = k\,[\,I_1(qR)\,K_0(qa) + K_1(qR)\,I_0(qa)\,] / \{\,I_1(qR)\,[\,k\,K_0(qb) + q\,K_1(qb)\,] + K_1(qR)\,[\,k\,I_0(qb) - q\,I_1(qb)\,]\}$$

and

$$G(s) = 2\,b\,k\,[\,I_1(qR)\,K_1(qb) - K_1(qR)\,I_1(qb)\,] / \{\,I_1(qR)\,[\,k\,K_0(qb) + q\,K_1(qb)\,] + K_1(qR)\,[\,k\,I_0(qb) - q\,I_1(qb)\,]\,L\,(R^2 - b^2)\,q\}$$

where $q = \sqrt{(s/D)}$. The temporal change of the associated process is determined by the Laplace inverted transform of the $\Phi(s)$.

By expanding the modified Bessel functions on the numerator and denominator by Allen's approximation method, we have

$$G(s) = \sum_{n=0}^{26} Nn\ q^{2n} / \sum_{n=0}^{28} Dn\ q^{2n}$$

The parameters were

ki : microscopic non specif association rate constant
>> $2\pi D1$ (diffusion control limited)
$= \lambda * K_{RD}$: $K_{RD} = 10^3$ /M

D = 6.5 10^{-7} cm^2 /s -- 5 10^{-7} cm^2 /s (Berg 1978)
l = length of a base pair (20 to 100 base pairs /L Berg 1981) distance between the binding sites

M = Dt/Ot. 50000 base paires for long chain
6700 base pairs for intrmediate chain
175 - 200 base pairs rod like chain)

Rc = 1.2 10^{-5} cm $= 0.6\,(L\,a^3)^{1/4}$
b = 5 10^{-7} cm L = 1.7/2 10^{-3} cm
$\alpha = \pi Rc/L$ $\beta = \pi b/L$ $\Gamma 1 = D1/l^2$
ν = 100/s - 1000/s - 10000/s
m = Lo/l > 25 (P 6945, A-16. M = 2m +1 , m =(M-1)/2)
λ : microscopic non specific dissociation rate constant
= 10^3/s (Berg 1978)
D1= 9 10^{-10} cm^2 /s - 3 10^{-10} cm^2 /s
Λ = 5 10^4 /s Λ $D1/(\Lambda l^2) = 100^2$
$l = (D1/(10^4\ \Lambda))^{1/2} = 10^{-9}$
L = 10^5 base pairs

where Ot : the total concentration of specific sites. M is the number of non specific sites on the DNA chain which include one operator. b: the

reaction radius (radius of DNA double stand). Λ : . intra domain non specific dissociation constant. Rc is a measure of the segment density in the DNA coil which is under the constraint of the relation

$$2\ \pi L\ Rc^2 = 4/3\ \pi\ rg^3$$

rg is the radius of rotation of DNA coil.

$rg^2 = 2\ L\ a\ /3$ where a is persistence length of the DNA.

3. Results.

Fig 2 shows the impulse response of $G(s)\ L\ (R^2 - q^2)\ /(2\ b\ k)$ at k= 0.0001. The computation has continued until t = 8* 10-3. The impulse response has oscillated significantly with several different frequency components. Fig 3 shows the impulse response at k=1.

Fig 2

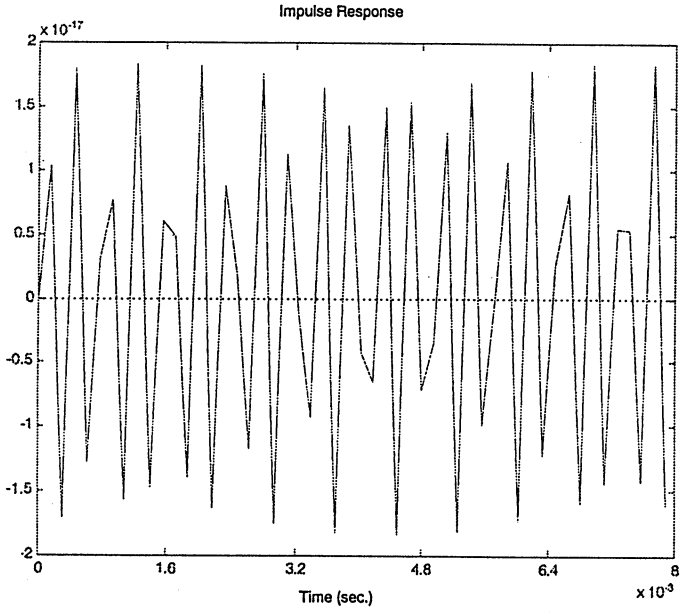
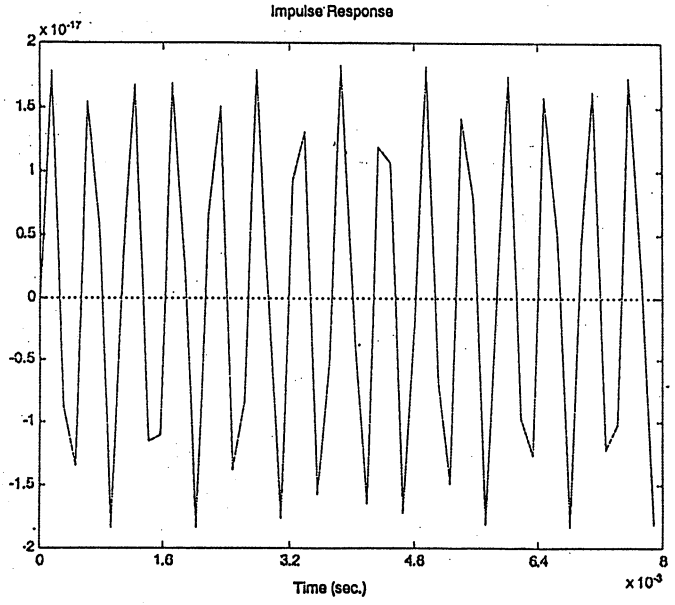


Fig 3



The amplitude of the oscillation seems to be smooth but still has considerable oscillational change.

These significant oscillation in the first time arrival probability of gene regulation protein particle indicates that at the onset of the gene expression, the kinetic behavior is rather chaotic rather than deterministic. This tendency could be observed when several other parameters have been altered. Hence at the initial phase of the gene regulation, the following process will be also probabilistic process.

4. Discussion.

The present computation based on the approximation process for modified Bessel functions. Hence, there may be errors which must have been augmented. The present computation, however, suggested that the regulation of gene expression is not a steady process but would be chaotic process.

5. Conclusion.

Gene control by repressor protein particle must be unstable and chaotic.

6. Reference.

1. Berg, O. BioPhy. Chemistry. vol 4. pp 367-381. 1976.
2. Berg O G., Winter R B and von Hippel P H. Diffusion-Driven mechanics of protein tanslocation on Nucleic acids. I. Models and theory. Biochemistry vol 20. pp 6929-6948. 1981.
3. Carslaw H.S and Jaeger J.C. Conduction of heat in solids. Oxford Univ., Press. 1947.
4. Richter P H and Eigen M. Biophys. Chem. vol 2. pp 255. 1974.
5. Winter R B and von Hippel P H. Diffusion-Driven mechanics of protein tanslocation on Nucleic acids. 2. The Escherichia coli Repressor-operator interaction; Eqilibrium measurements. Biochemistry vol 20. pp 6948-6960. 1981.
6. Winter R B and von Hippel P H. Diffusion-Driven mechanics of protein tanslocation on Nucleic acids. 3. The Escherichia coli Repressor-operator interaction; kinetics measurements. Biochemistry vol 20. pp 6961-6977. 1981.

APPENDIX 1.

We show the precise mathematical process for obtaining the r dependent part of the solution.

For r < a

$$w(r,s) = [\ I_1(qR)\ K_0(qa) + K_1(qR)\ I_0(qa)\ ] / \{2\ \pi D\ I_1(qR)\ [\ k\ K_0(qb) + q\ K_1(qb)\ ] + K_1(qR)\ [\ k\ I_0(qb) - q\ I_1(qb)\ ]\ * \{\ [\ k\ K_0(qb) + q\ K_1(qb)\ ]\ I_0(qr) - [\ k\ I_0(qb) - qI_1(qb)\ ]\ K_0(qr)\}\quad (q = \sqrt{s/D}\ )\quad \text{------A1}$$

The Laplace transformed z- independent flux $\psi(t)$ of particles onto the chain is

$$\phi(s) = 2\ \pi b D \left.\frac{\partial w}{\partial r}\right|_{(r=b)} = k\ \frac{[\ I_1(qR)\ K_0(qa) + K_1(qR)\ I_0(qa)\ ]}{\{\ I_1(qR)\ [\ k\ K_0(qb) + q\ K_1(qb)\ ] + K_1(qR)\ [\ k\ I_0(qb) - q\ I_1(qb)\ ]\ \}} \quad \text{----------A2}$$

The z- dependent flux in the equation (12) is equal to

$$F(z,z',t) = \psi(t)\ v(z,t) \quad \text{----------A3}$$

Hence the Laplace transform of F is

$$F(z,z',s) = \phi(s)/L + 2/L \sum_{n=1}^{\infty} \phi(s + D(n\pi/L)^2) \cos(n\pi z'/L)\cos(n\pi z/L) \quad \text{----------A4}$$

The Laplace transform of flux G(z,s) is calculated through the integration F(z,z',s).

$$G(s) = 2bk[I_1(qR)K_1(qb) - K_1(qR)I_1(qb)] / \{ I_1(qR)[k K_0(qb) + q K_1(qb)] + K_1(qR)[k I_0(qb) - q I_1(qb)] L (R^2 - b^2) q\} \quad \text{----------A5}$$

APPENDIX 2. Induction of equation A1.

Theorem

For the infinite hollow cylinder $a < r < b$, $0 < z < L$. r=a kept at temperature f(z). Radiation at the other surfaces into medium at zero

$$v = 2 \sum (\alpha_n \cos(\alpha_n z) + h \sin(\alpha_n z)) \phi(r;n) \int f(z)(\alpha_n \cos(\alpha_n z) + h \sin(\alpha_n z))dz / [(\alpha_n^2 + h^2) L + 2h] \phi(a;n)$$

where

$$\phi(r;n) = Io(r\alpha_n)[\alpha_n K1(b\alpha_n) - h Ko(b\alpha_n)] + Ko(r\alpha_n)[\alpha_n I1(b\alpha_n) + h Io(b\alpha_n)]$$

Proof

The basic differential equation of this system is

$$\partial^2 v/\partial r^2 + 1/r\, \partial v/\partial r = k\, \partial^2 v/\partial z^2$$

The boundary conditions are

$$v = f(z) \quad \text{at } r = a$$
$$\partial v/\partial r + h v = 0 \quad \text{at } r = b$$
$$-\partial v/\partial z + h v = 0 \quad \text{at } z = 0$$
$$\partial v/\partial z + h v = 0 \quad \text{at } z = L$$

The general form of the solution is

$$v = \sum [An\, Io(r\alpha_n) + Bn\, Ko(r\alpha_n)][Cn \cos(\alpha_n z) + Dn \sin(\alpha_n z)].$$

Putting $Cn/Dn = \alpha_n/h$,

$$v = \sum [An\, Io(r\alpha_n) + Bn\, Ko(r\alpha_n)] Cn [\cos(\alpha_n z) + h/\alpha_n \sin(\alpha_n z)].$$

Setting $An = An\, Cn$, $Bn = Bn\, Cn$ and $Xn = \cos(\alpha_n z) + h/\alpha_n \sin(\alpha_n z)$

$$v = \sum Xn [An\, Io(r\alpha_n) + Bn\, Ko(r\alpha_n)]$$

From the boundary condition (2) at r = b,

$$\partial v/\partial t = Xn [An\, \alpha_n I1(r\alpha_n) - Bn\, \alpha_n K1(r\alpha_n)]$$

Thus, we have

$$An\, \alpha_n I1(b\alpha_n) - Bn\, \alpha_n K1(b\alpha_n) + h[An\, Io(b\alpha_n) + Bn\, Ko(b\alpha_n)] = 0.$$

This is

$$An[\alpha_n I1(b\alpha_n) + h\, An\, Io(b\alpha_n)] = Bn[\alpha_n K1(b\alpha_n) - h\, Ko(b\alpha_n)].$$

For ths simple, we express this

$An\, C1 = Bn\, C2.$

Substituting $Bn = An\, C1/C2$ to v

$$v = \sum Xn\, An/C2\, [C2\, Io(r\alpha_n) + C1\, Ko(r\alpha_n)]$$

Putting

$$C2\, Io(r\alpha_n) + C1\, Ko(r\alpha_n) = [\alpha_n K1(b\alpha_n) - h Ko(b\alpha_n)] Io(r\alpha_n) + Ko(r\alpha_n)[\alpha_n I1(b\alpha_n) + h Io(b\alpha_n)] = \phi(r;n).$$

Then we have

$$v = \sum Xn\, An\, \phi(r;n)/C2.$$

Applying the boundary condition

$v = f(z)$ at $r = a$,

we have

$$f(z) = \sum Xn\, An\, \phi(a;n)/C2.$$

Multiplying Xm(z) on both side of this and integrate [0, L], we have

$$\int f(z) Xm(z)\, dz = \sum An\, \phi(a;n)/C2 \int Xn\, Xm\, dz$$

The right side of the equation is effective only for m=n

$$\int Xn^2\, dz = [(\alpha_m^2 + h^2) L + 2h] / (2\alpha_m^2)$$

Therefore

$$\int f(z) Xm(z)\, dz = An\, \phi(a;n)/C2\, [(\alpha_m^2 + h^2) L + 2h] / (2\alpha_m^2).$$

Thus, we have

$$An = 2\alpha_n^2\, C2 / \{[(\alpha_n^2 + h^2) L + 2h]^* \phi(a;n)\} \int f(z) Xn(z)\, dz$$

$$Bn = 2\alpha_n^2\, C1 / \{[(\alpha_n^2 + h^2) L + 2h]^* \phi(a;n)\} \int f(z) Xn(z)\, dz$$

Hence,

$$v = \sum Xn [An\, Io(r\alpha_n) + Bn\, Ko(r\alpha_n)] = 2\sum \alpha_n^2\, Xn^* [C2\, Io(r\alpha_n) + C1\, Ko(r\alpha_n)] / \{[(\alpha_n^2 + h^2) L + 2h]^* \phi(a;n)\} \int f(z) Xn(z)\, dz$$

Consequently, we have A1.

APPENDIX 3

$$qb = b^* \sqrt{s/d} = b^* 10^{-7} \sqrt{s}/(5^* 10^{-7})^{1/2} = \sqrt{s} * 2.68 * 10^{-3.5}$$

$$qR = R^* \sqrt{s/d} = b^* 10^{-5} \sqrt{s}/(5^* 10^{-7})^{1/2} = \sqrt{s} * 2.68 * 10^{-1.5}$$

Therefore, we can apply the Allen's approximation for I0(qb), I1(qR), K0(qb), K1(qR) and I1(qb). to satisfy $x \leqq 3.75$

$$\sqrt{s}^* 2.68 * 10^{-3.5} \leqq 3.75 \quad \text{Hence} \quad s \leqq 10^7 (3.75/2.68)^2$$

$$\sqrt{s}^* 2.68 * 10^{-1.5} \leqq 3.75 \quad \text{Hence} \quad s \leqq 10^3 (3.75/2.68)^2$$

Hence, we have sufficently large s

$$1)\ I0(qb) = \sum_{k=0}^{6} ak (qb/3.75)^{2k} = \sum_{k=0}^{6} ak\, q^{2k}$$

$$2)\ I1(qb) = qb \sum_{k=0}^{6} ak (qb/3.75)^{2k} = \sum_{k=0}^{6} bk\, q^{2k+1}$$

$$3)\ K0(qb) = -\log((qb)/2)^* I0(qb) + \sum_{k=0} ak(qb/2)^{2k}$$

$$5)\ K1(qb) = \log((qb)/2)^* I1(qb) + \sum_{k=0} ak(qb/2)^{2k}/b$$

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Extracting an independent component with a microphone array

Takashi IWAMOTO
Advanced Technology R&D Center
Mitsubishi Electric Corporation
Amagasaki, Hyogo 661-8661, Japan
E-mail: *iwataka@qua.crl.melco.co.jp*

Abstract

The author introduces a constructive scheme to extract an independent component with a microphone array. It consists of estimation of source directions based on mutual information between components extraction of a component based on estimated source directions.
Key words: independent component analysis, microphone array, mutual information, blind signal separation, blind signal deconvolution, landscape

1 Introduction

Considerable number of studies have been made on independent component analysis (ICA)[1]. In particular much work has been reported on blind signal separation (BSS)[2]. This is the task to extract indcpendent components from sensor signals $\mathbf{x}(t) = (x_1(t), \ldots, x_n(t))^T$ into which unknown source signals $\mathbf{s}(t) = (s_1(t), \ldots, s_n(t))^T$ have been superimposed by an unknown non-singular matrix $\mathbf{A} \in \mathbf{R}^{n\times n}$ as

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t).$$

BSS is attained by finding a separating matrix $\mathbf{W} \in \mathbf{R}^{n\times n}$ which makes each component of the output

$$\mathbf{y}(t) = \mathbf{W}\mathbf{x}(t)$$

statistically independent with respect to each other. Effective algorithms for BSS have been propoesed[3, 4, 5]. Amari et al[5] shows that minimization of mutual information between components plays a main role for extraction of independent components in common.

A more realistic task is blind signal deconvolution, where unknown source signals have been superimposed by an unknown matrix of convolutions kernels $\mathbf{A}(k) \in \mathbf{R}^{n\times n}$, $k \in \mathbf{Z}$ into sensor signals

$$\mathbf{x}(t) = \mathbf{A}(t) * \mathbf{s}(t)$$

($*$ is a convolution operator). Simple method is achieved by finding a matrix of convolution kernels $\mathbf{W}(k) \in \mathbf{R}^{n\times n}$, $k \in \mathbf{Z}$

$$\mathbf{y}(t) = \mathbf{W}(t) * \mathbf{x}(t).$$

However, generalization from instantaneous product to convolution brings about large degree of indeterminateness; each translation $s_i'(t) = c_i(t) * s_i(t)$, where $c_i(k)$, $(i = 1, 2, \ldots, n)$, $k \in \mathbf{Z}$ is arbitrary matrix of convolution kernel, is neutral with respect to independency.

The author introduces a constructive scheme to extract an independent component with a microphone array. It consists of estimation of source directions and extraction of components based on estimated source directions. It is not suitable for tasks where reflection or reverberation of source signals are not very important, however, still contains important applications.

2 Estimation of source directions

The author has introduced a scheme to estimate directions of sources[6]. It utilizes the mutual information between two components extracted by a microphone array. For the sake of simplicity, let us concentrate on an array consists of three microphone separated d in a line. One component is extracted by a delay-and-sum array[7]

$$f(t,\tau) = \frac{(x_1(t-\tau) + x_2(t) + x_3(t+\tau))}{3}.$$

This array has sensitivity in the direction $\theta(\tau)$, which satisfies the condition $\tau = d\sin\theta/c$ with the sonic velocity c. The other component is extracted by another

delay-and-sum array

$$r(t,\tau) = x_2(t) - f(t) = \frac{(-x_1(t-\tau) + 2x_2(t) - x_3(t+\tau))}{3},$$

which is insensitive in the direction $\theta(\tau)$. A plot of the mutual information is shown in Figure 1, where three sensor signals are generated from four signals emulating that sources are located in the direction of $\tau_1 = -27$, $\tau_2 = 0$, $\tau_3 = 27$, $\tau_4 = 54$ respectively. Relative minima of mutual information $I(\tau) = I(f(t,\tau), r(t,\tau))$ shows the direction of these sources.

Using characteristic resolution of a microphone array, as shown in Figure 1, we accelerate search of relative minima as described below.

Now we call a triplet (τ_1, τ_2, τ_3), with $\tau_1 < \tau_2 < \tau_3$, a convex segment of a function f if the condition $f(\tau_1) \geq f(\tau_2)$ and $f(\tau_3) \geq f(\tau_2)$ is satisfied. Let's two points τ_{12} and τ_{23} be midpoints as $\tau_{12} = (\tau_1 + \tau_2)/2$ and $\tau_{23} = (\tau_2 + \tau_3)/2$. We use the following proposition:

> Suppose that a triplet $T_1 = (\tau_1, \tau_2, \tau_3)$ is convex segment of a function f, there is at least one convex segment of the function f among the three subsegments: $T_{11} = (\tau_1, \tau_{12}, \tau_2)$, $T_{12} = (\tau_{12}, \tau_2, \tau_{23})$, and $T_{22} = (\tau_2, \tau_{23}, \tau_3)$.

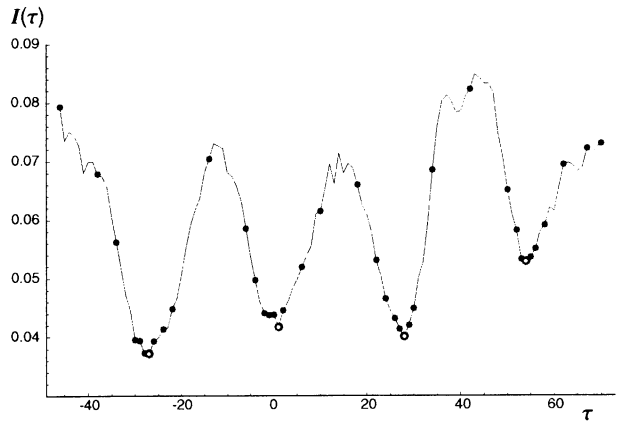


Figure 1: A plot of the mutual information against three sensor signals generated from four signals emulating that sources are located in the direction of $\tau_1 = -27$, $\tau_2 = 0$, $\tau_3 = 27$, $\tau_4 = 54$ respectively. Four open circles are marked at the relative minimums obtained by the introduced algorithm and closed circles are temporary pointers used by it.

It is sufficient to prove that the segment T_{12} is convex on the assumption that each segment T_{11} and T_{22} is not convex. From the supposition $\tau_1 \geq \tau_2$ holds. Hence from the assumption $\tau_{12} > \tau_2$ holds. Similarly $\tau_{23} > \tau_2$ holds.

Using this proposition, a algorithm is derived to search relative minima in a certain interval in the space of direction. At first, it check convexity of a triplet with a lather large span, say Δ. Once it finds a convex segment, it divides the segment into small ones recursively. Since the space of direction currently considered is discreet, a smallest convex segment, which cannot be divided any more, is identified quickly. In other words, this algorithm obtain relevant outline of landscape, relative minima, quickly.

This algorithm is realized by the following steps:

Step 1.
Let a step size Δ be a slightly smaller number than a half of the resolution determined by the configuration of microphones.

Step 2.
Initialize the pointer τ_1 to be the smallest number of the target region and pointers τ_2 and τ_3 as $\tau_2 = \tau_1 + \Delta$, $\tau_3 = \tau_2 + \Delta$.

Step 3.
While the current triplet (τ_1, τ_2, τ_3) remains in the target region
do Step 5 and increment of the current triplet (τ_1, τ_2, τ_3) by Δ.

Step 4.
Out put each memorized relative minimum and **exit**.

Step 5.
If the current triplet (τ_1, τ_2, τ_3) is convex
then If the midpoint $\tau_{12} = (\tau_1 + \tau_2)/2$ and $\tau_{23} = (\tau_2 + \tau_3)/2$ are elements of the current triplet (τ_1, τ_2, τ_3)
then memorize the $(\tau_2, I(\tau_2))$ as a relative minimum
else for $(\tau_1, \tau_2, \tau_3) = (\tau_1, \tau_{12}, \tau_2)$, $(\tau_{12}, \tau_2, \tau_{23})$, and $(\tau_2, \tau_{23}, \tau_3)$ apply Step 5.

In Figure 1, pointers used in search of relative minima are also shown as closed circles.

3 Extraction of a component

The delay-and-sum array $f(t, \tau)$ works even for extraction of a component from sensor signals whose number is smaller than that of source signals. However, when the number of sensor signals are greater than or equal to that of source signals, better arrays are designed to extract a component. Let us concentrate on source signals $s_1(t)$, $s_2(t)$, $s_3(t)$ sampled at 22KHz for 5 second with 16bit depth (Figure 3).

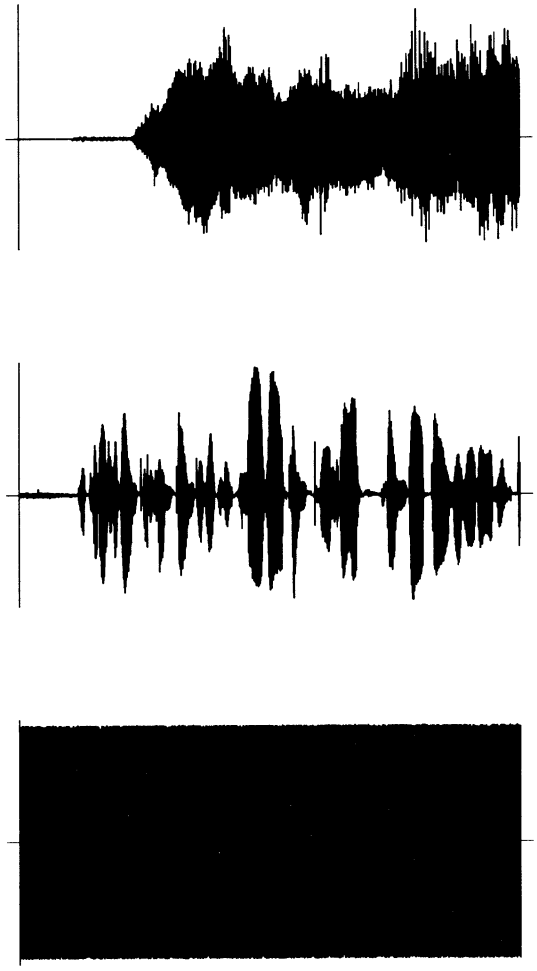


Figure 2: Three source signals. Each source is sampled for 5 second at 22KHz with 16bit depth.

Three sensor signals are generated from these source signals emulating that sources are located in the direction of $\tau_1 = -20$, $\tau_2 = 0$, $\tau_3 = 20$ respectively (Figure 4).

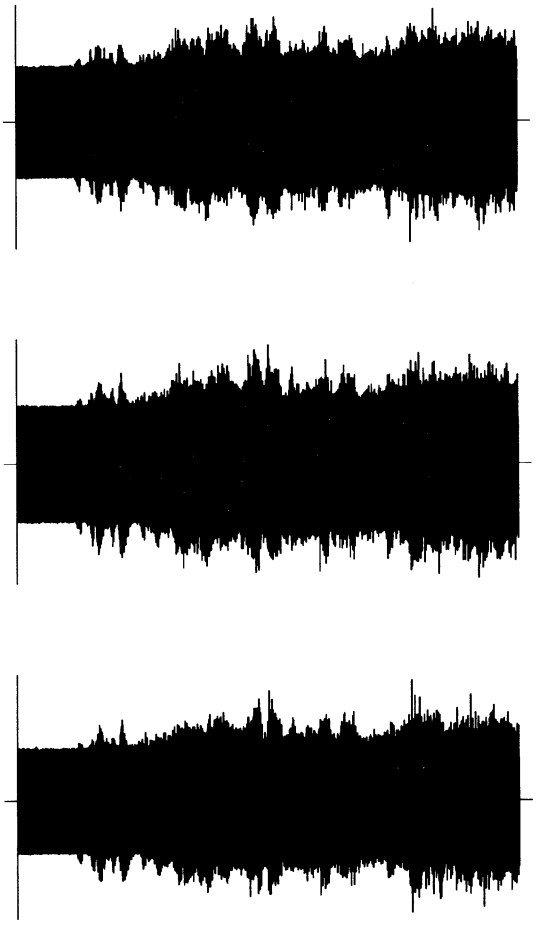


Figure 3: Simulated three sensor signals.



Figure 4: The extracted component from the sensor signals shown in Figure 3.

A delay-and-sum array that has insensitivity in the directions τ_1, $\tau_3$is realized by

$$x_1(\frac{-\tau_1 - \tau_3}{2}) + x_3(\frac{\tau_1 + \tau_3}{2}) - x_2(\frac{\tau_1 - \tau_3}{2}) - x_2(\frac{-\tau_1 + \tau_3}{2}).$$

The extract component by this array is shown in Figure 5.

4 Summary

The author introduces a constructive scheme to extract an independent component with a microphone array. It consists of estimation of source directions and extraction of components based on estimated source directions. The estimation is accelerated by segmentation division of segments. The extraction is shown to work effective for tasks where reflection or reverberation of source signals are not very important.

Acknowledgements

The author is grateful to Mr. Ikeda and Mr. Takeda for data sampling and technical discussion.

References

[1] A. Hyvarinen, "Survey on Independent Component Analysis," *Neural Computing Surveys*, Vol. 2, pp. 94-128, 1999.

[2] C. Jutten and J. Herault, "Blind separation of sources, Part I: An adaptive algorithm based on neuromimetic architecture," *Signal Processing*, Vol. 24, pp. 1-10, 1991.

[3] P. Comon, "Independent component analysis, a new concept?" *Signal Processing*, Vol. 36, pp.287-314, 1994.

[4] A. J. Bell and T. J. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, Vol. 7, pp.1129-1159, 1995.

[5] S. Amari, A. Cichocki, and H. H. Yang, "A new learning algorithm for blind signal separation," In *Advances in Neural Information Processing Systems 8*, MIT Press, pp. 757-763, 1996.

[6] T. Iwamoto, "Estimating an independent component based on the mutual information against the residual component," In M. Sugisaka and H. Tanaka (ed) *Proceedings of the Fourth International Symposium on Artificial Life and Robotics (AROB 4th)*, Beppu, Oita, Japan, pp. 58-61, 1999.

[7] S. U. Pillai, *Array signal processing*, Springer-Verlag, 1989.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

H2 Control Strategy for Ion Channels on the Biological Membrane.

H, Hirayama., N, Kitagawa., Y*. Okita and T**. Kazui.

Asahikawa medical college. 1-1. Higashi 2-1. Midorigaoka Asahikawa city
hirayama @ asahikawa-med.ac.jp *Shizuoka university. ** Hamamatsu medical college.

Abstract

The present paper introduce H2 control principle for Sodium channel gating system on the cellular membrane. The Sodium ion channel is composed of four equivalent subunits which surrounds the pore through which ion current.Each subunit is composed of 6 cylindrical helixes S1 to S6. The opening and closing of the channel is explained by helical sliding model or patch ball. The multiple state was expressed by linear differential equations. By the Riccati operator, we obtained the controller that minimizes the functional space of the closed loop transfer function. The present paper will be available for evaluating the of Sodium channel gating system.
key words : H2 control. Sodium channel gating. Helical subunits. Biological membrane information processing.

1. Introduction.

Bio information is transmitted through discrete systems under some stochastic processes. Neuro chemical information transmission is the typical example of such systems. With the technical advances for measuring minute composite currents through the biological membrane, detailed mechanisms have been disclosed for several ion channel gating system as an biological information transmission. The ion channel systems can be categorized to ligand bound gating and membrane voltage dependent gating systems. The ligand type opens its channel pore by the molecular conformational changes in the channel site through binding the agnoist which molecular constructure is complemental to that of the channel receptor site. The voltage sensitive channel open and close its pore depending on the potential gap across the membrane. These two systems constitute the ultimate functional unit for molecular information processing systems. These mechanism when disclosed precisely can contributed to create artificial bio membrane and information transmission systems.

In the present study, we introduce application of H2 control for Na channel gating system as the voltage dependent membrane gating system.

Fig 1

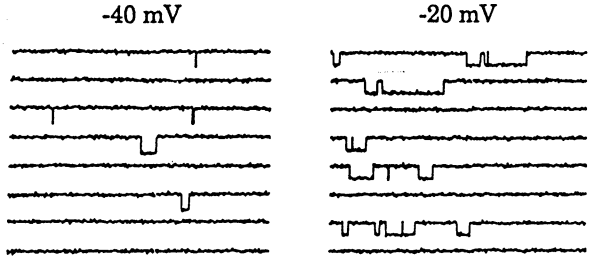


Fig 2-a

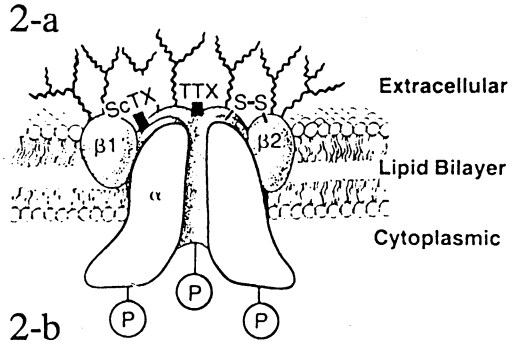


Fig 2-b

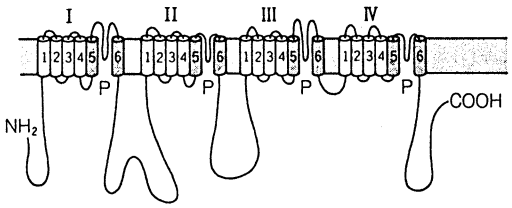


Fig 2-c

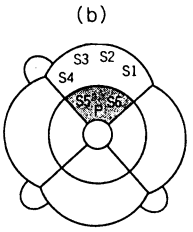


Fig 2-d

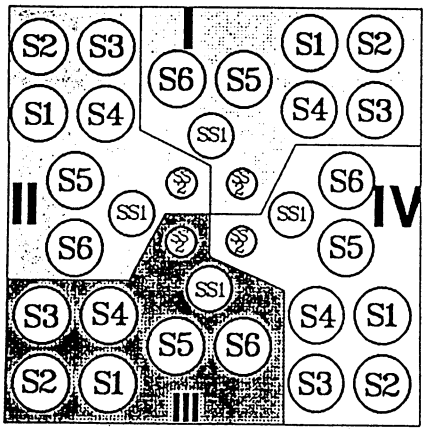


Fig 2-a. 3 dimensional illustration of molecular constitutional structure of Na channel.
Fig 2-b. Transversal view of membrane spaning regions.
Fig 2-c. Top view of the Na channel surrounded by four identical subunits.
Fig 2-d. Precise picture of the top view of Fig 2-c.

2. Physiological background.

Before proceed to mathematical process and H2 control, we give some explanations for important properties of Sodium (Na) gating channel.

Fig 1 shows membrane current through a single Na channel. The down ward square pulse waves are the inward current into the cell during which the channel has been open. The opening duration are interrupted by short closure. Depolarizing the membrane potential from -40 mV to - 20 mV increased the frequency of channel opening.

Fig 2-a is a schematic illustration of molecular conformation of Na channel subunits. P is phosphate. α and β are constitutional subunits composed of polypeptid. Fig 2-b is the transversal view of the Na channel. The channel hole is surrounded by four homologous subunits each of which also is composed of 6 different amino acid helixes with additional small fractions of amino acid. The S5 and S6 divisions of these subunits face to the internal surface of the channel pore. They are denoted by shaded ares. Fig 2-c is the top view of the channel pore which clearly show the constitution of the hole. Fig 2-d is a detailed schema of the top view of the channel pores. Putative positions of 8 membrane-associated regions are indicated within each repeat subunits. Pore for ionic permeation is at hub of structure.

For minute molecular structure, biochemists have discovered structural change of these helixes. Fig 3 shows three dimensional structure of the S4 helix domain which each constitutes the subunit. Each subunits is composed of six cylindrical circular helical amino axis strand which structure is complex. The composing amino acid sequence has already been determined. The diameter of each cylinder is 5 A. The regular α helix is composed of 20 amino acids. The planar peptide bonds were drawn. The positive charges on every third side chain induce regions of polarization around them which is indicated by hemisphere. Such conformational change form a left hand screening helix with 27 A per turn.

There are several inconsistency for channel opening and closing. We present representative two molecular mechanisms for channel opening and closing.

Fig 3

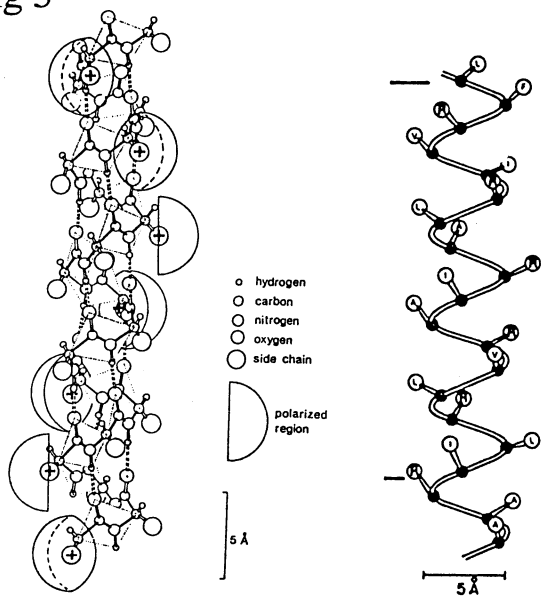


Fig 5

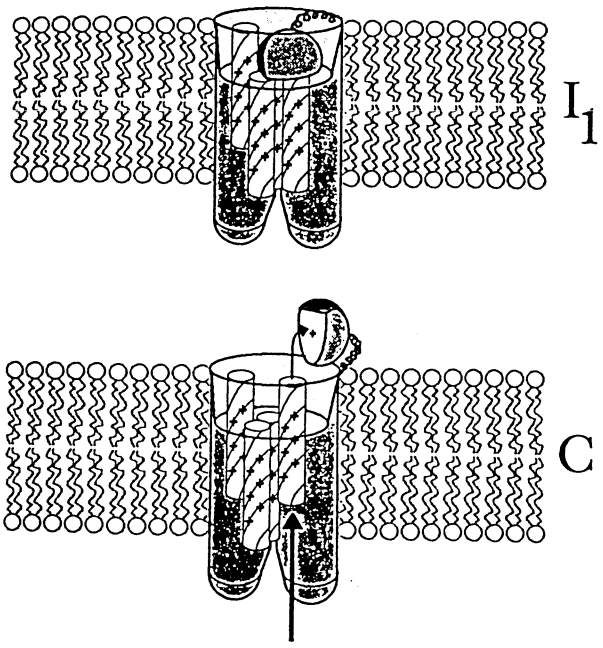


Fig4 shows the spiral path made by the positively charged arginine residues in the form of a ribbon wrapped around the central core of the helix. The S4 helix domain is driven to move helical swirly by the membrane depolarization. The trans membrane S4 helix is represented by a cylinder with spiral ribbon of positive charges. At resting membrane state, all positively charged residues are paired with fixed negative charges on other trans membraneous segments of the channel. The trans membraneous segment is held in that position by the negative internal membrane potential. Activation of the membrane through depolarization reduces the force holding the positive charges in their inward position. The S4 helix domain is undergo a screw-like swirl motion through rotating 60 deg and a n outward displacement of 5 A. This movement leaves an unpaired negative charge on the inward surface of the membrane. This configuration reveals an unpaired positive charge on the out ward surface

Fig 4

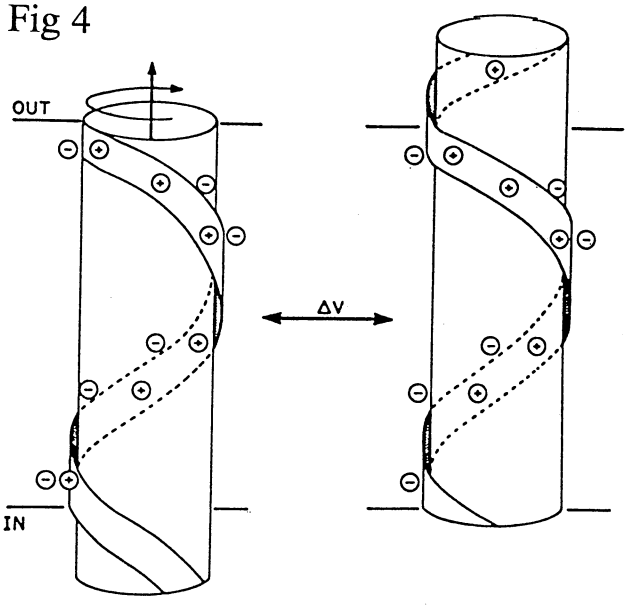
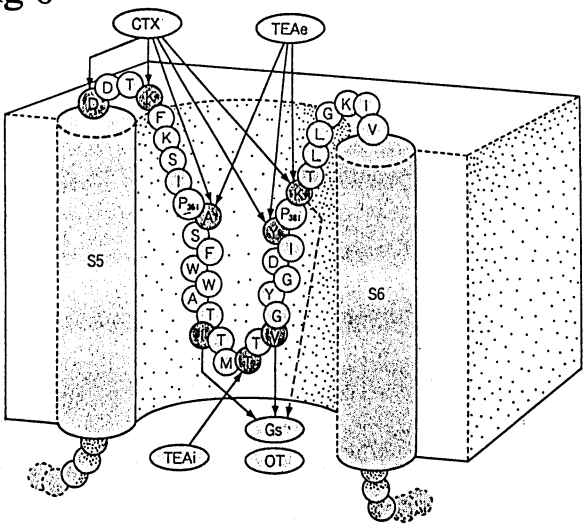


Fig 6



to give a net charge transfer of +1. (Catterall. W.A. Trends. Neurosci. vol 9. pp 7-10. 1986.)

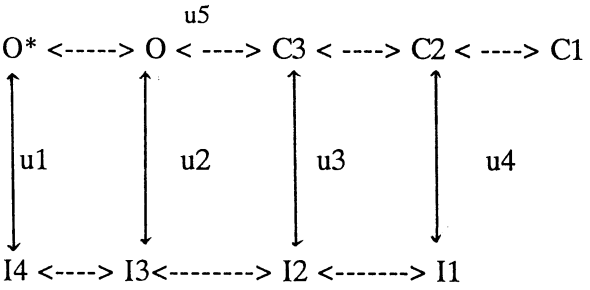
Fig5 shows model for membrane voltage-dependent recovery from inactivation. After strong depolarization, all four activated gates shown as charged trans membrane cylinder are in their open position. The inactivation particle occulted the pore. After repolarization, one activation gate that is not immobilized by inactivation particle turns to its closed conformation and giving rise to non immobilized fraction of return gating current. Subsequent recovery must wait either for spontaneous exist of inactivation particle or one of remaining activation gates to obtain sufficient energy to push out inactivation particle shown at the lower picture. This process depends on energy impaired to activation gates via membrane potential. Rate of recovery thus gains voltage dependence. (Patalak J. Physiological Reviews. vol 71. pp 1047-1080. 1991.)

In the last, we give a schema for minute structure of the inner surface of the channel pore surrounded by S5 and S6 particularly in relation to H5 division.
Fig 6 shows internal surface of the channel pore. S5 and S6 are the amino acid sequence cylinder. CTX and TEA (tetra ethyl ammonium) are the channel blockers that act specifically on the arrowhead position. This small region called H5 division determined the important properties of the channel including the sensitivities to the channel blockers such as CTX and TEA, the kinetic velocity for channel opening and closing and permeability of ion. (Sogabe M. Ion Channel in New Biophysics. No5. Kyouritu Pub. pp 75. 1997).

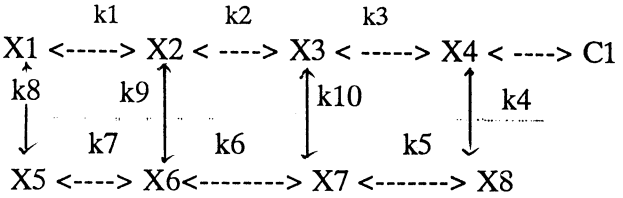
Fig 7. Cartoon representation of kinetic schemes used to model Na channel gating kinetics. The basic foundation was proposed by Hodgkin and Huxley. The resting state is at the upper left and the inactivated state that is reached after a period of strong depolarization is at the lower right. The one open state is at the upper right.

These sequential kinetics of multiple conformational changes in the channel states are represented as

Fig 8.

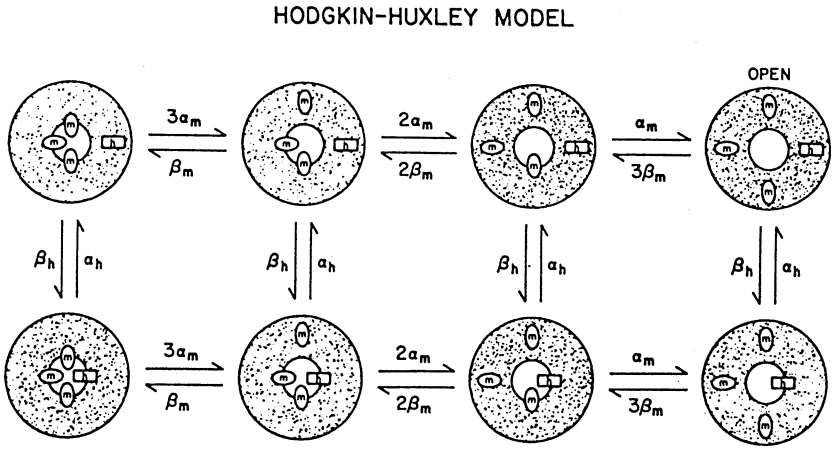


where O indicates open state, C closed states and I inactivated states. O* indicates the second brief opening state which rapidly transits to complete open state. C2 describes S1 domain is activated but closed. C3 for S1 and S2 domains are activated but closed. O indicates S1. S2 and S3 domains are activated by closed. o* indicates that S1, S2, S3 and S4 are activated. I4 corresponds to O* but inactivated. I3 corresponds to O but inactivated, I2 corresponds to C3 but inactivated and I1 to C2 inactivated. Un are controlling membrane voltage acting on those specific conformations of the channel gating structures. By setting these channel gating states to the state variables, the above scheme is represented as following



where kn are th rate constants.

Fig 7



3. Mathematical process for H2 control.

Above system can be expressed by eight linear differential equations by the mass action law as

$$\partial X1/\partial t = k1\,X2 + k8\,X5 - (k\text{-}1 + k\text{-}8)\,X1 + b1\,U1 \quad \text{----------(1)}$$

$$\partial X2/\partial t = k\text{-}1\,X1 + k2\,X3 + k9\,X6 - (k1 + k\text{-}2 + k\text{-}9)\,X2 + b2\,U2 + b3\,U5 \quad \text{-----(2)}$$

$$\partial X3/\partial t = k\text{-}2\,X2 + k3\,X4 + k10\,X7 - (k2 + k\text{-}3 + k\text{-}1)\,X3 + b5\,U5 + b4\,U3 \quad \text{-----(3)}$$

$$\partial X4/\partial t = k\text{-}3\,X2 + k4\,X8 - (k3 + k\text{-}4)\,X4 + b6\,U4 \quad \text{-----(4)}$$

$$\partial X5/\partial t = k\text{-}8\,X1 + k7\,X6 - (k8 + k\text{-}7)\,X5 + b7\,U1 \quad \text{-----(5)}$$

$$\partial X6/\partial t = k\text{-}9\,X2 + k\text{-}7\,X5 + k6\,X7 - (k9 + k7 + k\text{-}6)\,X6 + b8\,U2 \quad \text{-----(6)}$$

$$\partial X7/\partial t = k\text{-}10\,X3 + k\text{-}6\,X6 + k5\,X8 - (k10 + k6 + k\text{-}5)\,X7 + b9\,U3 \quad \text{-----(7)}$$

$$\partial X8/\partial t = k\text{-}5\,X7 + k\text{-}4\,X4 - (k5 + k4)\,X8 + b10\,U4 \quad \text{-----(8)}$$

where Xn indicates the concentration per unit membrane area of corresponding channel state. With the conservation law

$$\Sigma Xn = Et \quad \text{--------(9)}$$

those equations were reeduced to seven differential equatins which can be expressed as

$$\mathbf{X}' = A\mathbf{X} + B\mathbf{U} \quad \text{--------(10)}$$

where matrixes are

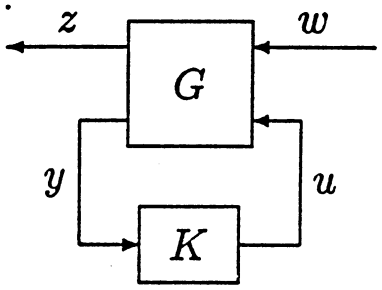
$$A = \begin{bmatrix} 0 & a12 & 0 & 0 & a15 & 0 & 0 \\ a21 & 0 & a23 & 0 & a25 & a26 & 0 \\ 0 & a32 & 0 & a34 & 0 & 0 & a37 \\ a41 & a42 & a43 & a44 & a45 & a46 & a47 \\ a51 & 0 & 0 & 0 & 0 & a56 & 0 \\ 0 & a67 & 0 & 0 & a65 & 0 & a67 \\ a71 & a72 & a73 & a74 & a75 & a76 & a77 \end{bmatrix}$$

$$B = \begin{bmatrix} b1 & 0 & 0 & 0 & 0 \\ 0 & b2 & 0 & 0 & b3 \\ 0 & 0 & b4 & 0 & b5 \\ 0 & 0 & 0 & b6 & 0 \\ b7 & 0 & 0 & 0 & 0 \\ 0 & b8 & 0 & 0 & 0 \\ 0 & 0 & b9 & 0 & 0 \end{bmatrix}$$

The matrix for noise wn is

$$B1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Fig 9.



$$G(s) = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & 0 & D_{12} \\ C_2 & D_{21} & 0 \end{array}\right].$$

$z = C1\,X + D12\,u$ $\qquad$ $Y = C2\,X + D12\,w$

$$C1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & q2 & 0 & 0 & 0 & 0 & 0 \\ q1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & q3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & q4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & q5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & q6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & q7 \end{bmatrix} \qquad C2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$D21 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where G (Fig 9) is the plant with two sets of inputs, exogenous input w including disturbance noises and control input u. The out put from G is the measured outputs y and the regulated out put z. K is the controller. In the present paper, K is the membrane potential. z and Y are expressed by above equations with coefficient matrixes. The H2 control is that Find admissible controllers K(s) such that II Tzw II_2 is minimized where Tzw is the closed loop transfer function from w to z.

3. Results.

Fig 10 shows the transient changes of the amount of individual channel state per unit membrane area. The present analysis is just an introduction of application of the Robust control theory for neuro chemical transmission. Strong comparison is required with biological experiments.

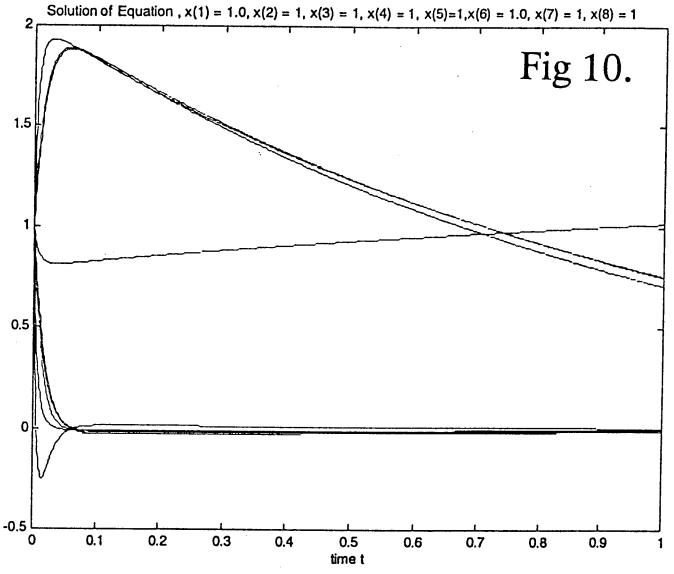


Fig 10.

4. Conclusion.

Modeling of Sodium channel under the H2 control principle will be available for evaluating the neuro chemical signal information processing.

5. References.

1. Benndorf K. Eur. Biophys. J vol 17. pp 257-271. 1989.
2. Catterall. W.A. Trends. Neurosci. vol 9. pp 7-10. 1986
3. French R J. Ann. Rev. Biophysvol 12. 319-356. 1983.
4. Guy H. In. Monovalent cations in Biological system. ed by Pasternark. CRC. 19909. pp 31-58. 1990.
5. Patalak J. Phys Reviews. vol 71. pp 1047-1080. 1991.
6.Sogabe M. Ion Channel in New Biophysics. No5. Kyouritu Pub. pp 75. 1997

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

From Human Error to "Agent Error"

Laurent Chaudron and Nicolas Maille
ONERA CERT 2 avenue Edouard Belin
BP 4025 - 31055 Toulouse Cedex 04 - FRANCE
Phone: +33 5 62 25 26 55 Fax: +33 5 62 25 25 64
{chaudron, maille}@cert.fr

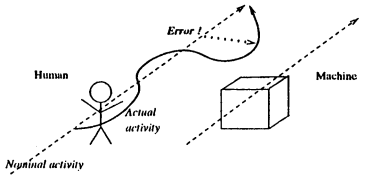
Topics: Artificial Intelligence, Cognitive Science.
Key words: Knowledge Representation, Multiagent Systems, Error Modeling.

Meta-abstract

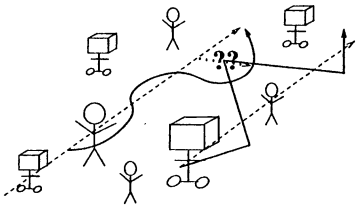
In this paper we describe a formal model for knowledge based error representation dedicated to mixed systems, i.e. complex systems in which human and machines cooperate with each other. Our contribution consists in: i) a conceptual shaping of the "agent error" conjecture, ii) the first steps towards a formal description of a generic "agent error" model.

General aspects of agent error

In the aeronautics area it is frequently stated that *X% of incidents/accidents are due to "human errors"*. Such a formula is intrinsically ambiguous to analyze especialy in the case of the new sophisticated cooperative systems in which the performances of the artificial agents (aircrafts, drones, autonomous vehicles, smart sensors, and more generally all intelligent information support subsystems: ATC...) are becoming closer to those of human beings.
In a "classical" human-machine interaction, an error is considered [5] as a deviation from a "nominal" activity [1]. In



such a simplified configuration, it is clear (even if not easy) to identify and to discriminate human errors and machine failures. The situation is different if the machines are more sophisticated; humans and machines must then be considered as heterogenous agents of a global multiagent system in which knowledge processing and exchanges are the main cooperation means. Here it appears more difficult to distinguish between purely human agents errors and artificial artificial agents' ones as both categories manage the same kind of elements of information. Moreover, the dynamics of the exchanges induces knowledge interlacings.



Consequently, it appears necessary to design a generic model allowing knowledge deviations to be captured so as to to shift form the concept of human error to the *agent error* one.

Theoretical aspects of agent error

From the information processing point of view, the informal notion of deviation to a nominal activity must get a corresponding definition in terms of a strutured difference betwen two pieces of knowledge. When numerical data are considered, various tools are available to deal with deviations and approximations. The problem is quite different when the data are symbolic, in so far as it becomes impossible to consider statements like "formula ζ is an approximation of formula ξ to the nearest 25%" or to cope with equations like: "$\psi = \varphi - \delta\varphi$". As a first step, a formal model, the *Cube* model has been designed. A cube is a finite set of first order logic properties; the basic symbolic fusion operators (supremum, infimum, partial order) are defined and implemented in CLP (Constrained Logic Programming). The set of cubes is equiped with a lattice structure [2].
For example, in an experience feedback project [3] dedicated to flight incident analysis and human error understanding, the aircrew activity and the subsystems behaviors are described in terms of sequences of cubes. The cube model allows to compare the nominal activies and the observed ones (mainly deduced from the flight parameters recordings) so as to point out the deviations and the discrepancies.
Thus, agents errors are measurable in terms of symbolic knowledge differences.

Conclusion

Our approach aims at providing correct definitions for naive notions related to error management in complex mixed systems. In a first step, the Cube model gives a strong algebraic structure allowing to consider both human and artificial errors in terms of knowledge deviations. The enrichment of this "agent error" model is currently under study (in particular the enhancement of the performances of the agents due to error management [4]).

References

[1] R. Amalberti and L. Wioland. Human error in aviation. In *Intl. Conf. on Aviation Safety*, Roterdam, The Netherlands, August 1997.

[2] L. Chaudron, C. Cossart, N. Maille, and C Tessier. A purely symbolic model for dynamic scene interpretation. *International Journal on Artificial Intelligence Tools*, 6(4):635–664, 1997.

[3] L. Chaudron, F. Dehais, P. Le Blaye, and L. Wioland. Human activity modelling for flight analysis. In *Proceedings of HCP'99, Intl. Conf. on Human Centered Processes*, Brest, France, 22-24 September 1999.

[4] L. Chaudron, J. Erceau, and C. Tessier. Cultural differences? an opportunity for a mutual knowledge enhancement. In *Proceedings of PRICAI98, Pacific Rim Intl. Conf. on Artificial Intelligence, Workshop "Issues of Cross Cultural Communication"*, Singapore, 22-27 Nov. 1998.

[5] J. Reason. *Human Error.* Cambridge University Press, 1990.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Why people play: Artificial lives acquiring play-instinct to stabilize productivity

Shinichi Tamura
Division of Functional Diagnostic Imaging, Osaka University Medical School
D11, 2-2 Yamadaoka, Suita City, Osaka 565-0871, JAPAN
Phone: +81-6-879-3560 Fax: +81-6-879-3569 tamuras@image.med.osaka-u.ac.jp

Shoji Inabayashi Waichi Hayakawa Takahiro Yokouchi Youji Katou
Pacific Systems Corp.

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 varaiety in a species decreased its species will be extinguished. Thus we show some simulation results, e.g., in the world where more variety of abilities are requested in the plays the performance of the whole world calculated only from job tasks becomes steady and can avoid the risk of extinction of the species. This is the effect of the play.
Keywords: Artificial life, Artificial mind, Play, Genetic algorithm, Human ability

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 refered 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 evolution 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 varaiety in a species decreased its species will be extinguished.

This paper shows an attempt to generate an artificial mind which has two aspects of "diligent to his job" and "apt to run to pleasure." We show the latter does not have a negative meaning but also has a positive meaning to cope with variously changing environment as a group and stabilize the world productivity.

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=[interest-in-fishing], etc. The set of such abilities is defined as

$$A = \{a_1, a_2, \ldots, a_N\}. \quad \text{(Ability set)} \quad (1)$$

Null element ϕ 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 Et at time t is expressed as a set

$$Et = \{T_{t1}, \ldots, T_{tm}\} \; ; \; t=1,2,\ldots \quad \text{(Environment)} \quad (2)$$

of requested tasks in the environment, where tasks are, e.g.,

$T_{\text{old-times 1}}$ =[protect-from-wild-animals],
$T_{\text{recent-ages 1}}$ =[publish-book], $T_{\text{recent-ages 2}}$ =[open-a-store],
$T_{\text{recent-ages 3}}$ =[negotiate-with-other-company],
$T_{\text{today 1}}$ =[set-computer-business-up].

The environment changes time to time.

2.3 Task

Each task Ti is expressed by a set of requested abilities to accomplish it as:

$$T_i=\{a_{i1}, a_{i2},...,a_{i\,n_i}\}\ ;\quad i=1,...,N_T. \qquad \text{(Task)} \qquad (3)$$

For example,

T_1(=task of [protect-from-wild-animals])
= {quick-motion, cool-mind, good ear,...}.

T_2(=task of [political-action])
= {fair-spoken, planning-ability, ...}

2.4 Play

On the other hand, there are a set Pt of all hobbies including plays, self-development, or volunteer activities, etc. to which people begin or join voluntarily at time t. That is

$$P_t=\{R_{t1},...,R_{t\,m_t}\}\ ;\quad t=1,2,... \quad \text{(Play set)} \qquad (4)$$

For example,

Pt={soccer, go-game, marathon, reading, travel, fishing, volunteer-activity, ...}

Each play also requests set of abilities:

$$R_i=\{a_{i1}, a_{i2},...,a_{i\,m_i}\}\ ;\quad i=1,...,N_R. \quad \text{(Play)} \qquad (5)$$

For example,

R_1(=play of [fishing])
={interest-in-fishing, driving car, patient, ...}

R_2(=play of [go-game])
={interest-in-go-game, pattern recognition, spatial reasoning, tactical sense,...}

Abilities of [interest-in-*] is always necessary to begin the play of *. The set Pt of plays may 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 Ik has his abilities and their skills:

$$I_k=\{a_{k1},...,a_{k\,n_I};\ s_{k1},...,s_{k\,n_I}\ ;\ p_k\}\ ;\quad k=1,2,...,N_I, \qquad \text{(Individual)} \qquad (6)$$

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 may be divided among them, or only the most fitted one is selected as shown in 3.4.

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

$$s_{ki}=0.5+\sum_t [\ \alpha_T \times (1-p_k) + \alpha_R \times p_k\ ] \qquad \text{(Skill)} \qquad (7)$$

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. α_T takes 1 when the ability a_{ki} is used in the task, and 0 when not used. α_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 Et is given, an individual most fitted to the task (refer 3.3) among randomly selected unassigned Ne (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

$$\min[\ 1,\ b\times p_k];\quad b \geqq 1 \text{ (e.g., b=2)}. \qquad (8)$$

Then the skills used implicitly as well as explicitly in the play will be incleased as much.

3.3 Fitness to task

The fitness of the individual $I_k=\{a_{k1},...,a_{k\,n_I};\ s_{k1},...,s_{k\,n_I}\ ;\ p_k\}$ to the task $T_i=\{a_{i1}, a_{i2},...,a_{i\,n_i}\}$ is given by sum of the skills used in the task as

$$f_{ik} = (1-p_k)\times \sum_{j=1}^{n_i} s_{kl} \qquad \text{(Fitness)} \qquad (9)$$

such that $a_{ij}=a_{kl}$ for some l in $\{1,2,...,n_I\}$.

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_{kl} 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 Ri 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

$$Q = \sum_{i=1}^{N_t} f_i^* \qquad \text{(Performance)} \qquad (10)$$

where f_i^* is f_{ik} of an individual Ik who is practically assigned to the task Ti. 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 ($\{a_{k1},...,a_{k\ nl}$; $p_k\}$) 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 interpretted 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. Simulation and experimental results

We have developed a simulation software.

The initial parameters of the simulation are N_I=120, N_T=100, n_i=16, N_R=64, m_i=0-64, b=2, and rate of mutation is 0.05. Zone of the ability code number more than 64 is allocated to the abilities of {interest-in-*}.

Fig.2 shows a behavior of the world according to the generation lapse with 64 kinds of less complicated plays each of which requires no ability. In Fig.2(a), the vertical axis represents the code number of the ability which is expressed by seven bits in the gene. The environment requestes abilities between two lines (i) changing stepwise according to the generations. Then, various abilities (ii) in the group of the artificial lives become to appear. When the generated ability distribution as a result of the evolution matches well to the requested abilities the performance (iii) of the world becomes high. When the environment changes suddenly as in the 1000-th, 1500-th, and 3000-th generation, the performance of the world becomes almost zero and have a risk of extinction of the species. The curve (iv) shows the variance of the ability distribution (ii). We can see when the ability distribution (ii) matches well with small variance, the performance becomes very high as in between the 2000-th and 3000-th generations. Fig.3 shows the behavior of the world with 64 kinds of highly complicated plays each of which requires 64 abilities. We can see that the performance of the world is fairly steady compared with Fig.2. The variance curve is also steady than Fig.2.

We can see that in the world where more variety of abilities are requested in the plays, or various kinds of plays are possible and popular as in the advanced countries, the performance of the whole world in spite of calculated only from tasks becomes more stable. This is the effect of the play.

Generally speaking, the simulation of GA is not always stable, but sometimes it happens that even when the performance is growing up steadily the evolution becomes bad suddenly. This is partly because of the limited number of individuals in the simulation.

Though we have not tried fully yet, 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 for its possible changes. Some part of such abilities embedded into an individual will appear as plays or hobbies. Thus, 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 become to appear. As a result, three types of individuals will be generated. The first type of the individual is fitted to his present task, and this type of individual is happy only in his job. The second one is happy in his play rather than in his job. The third one is their intermediate type. Essentially it may be nonsense to discriminate the abilities for the job tasks and for the plays, although in this paper we discriminated them for convenience of the simulation.

The present environment requests not the whole abilities of the group of humans but only the part. The unrequested abilities will appear as the play in the hobby world and prepare the next chance to be requested from the environment by being trained in the play.

5. Conclusion

In this paper, we have proposed a world model of a group of artificial lives capable of coping with various environment, where individuals with mind of plays or hobbies are generated. We have shown in a simulation that in the world where more variety of abilities are requested in the plays the performance of the whole worlds in spite of calculated only from tasks becomes stable. This is the effect of the play. Thus we may be able to say that we could confirm that the role of plays will be partly some kind of training for implicit or explicit abilities which will be useful for unexpected changes of the environment.

Though the simulation is in progress, further improvement along the following items will be necessary:

(1) The system behavior is rather dependent on how the environment changes in speed and range. Therefore,

we must investigate what type of changes will generate what type of artificial lives.
(2) The present model of the task is flat of only a personal job level. It might be needed to construct a task system with a hierarchical structure such as company composed of individuals.
(3) It may be difficult in the real world to discriminate the job tasks as the duty one and the hobbies as the voluntary one with pleasure.
(4) In the real world, evaluation of the world is made not only by the job tasks but by including the plays.

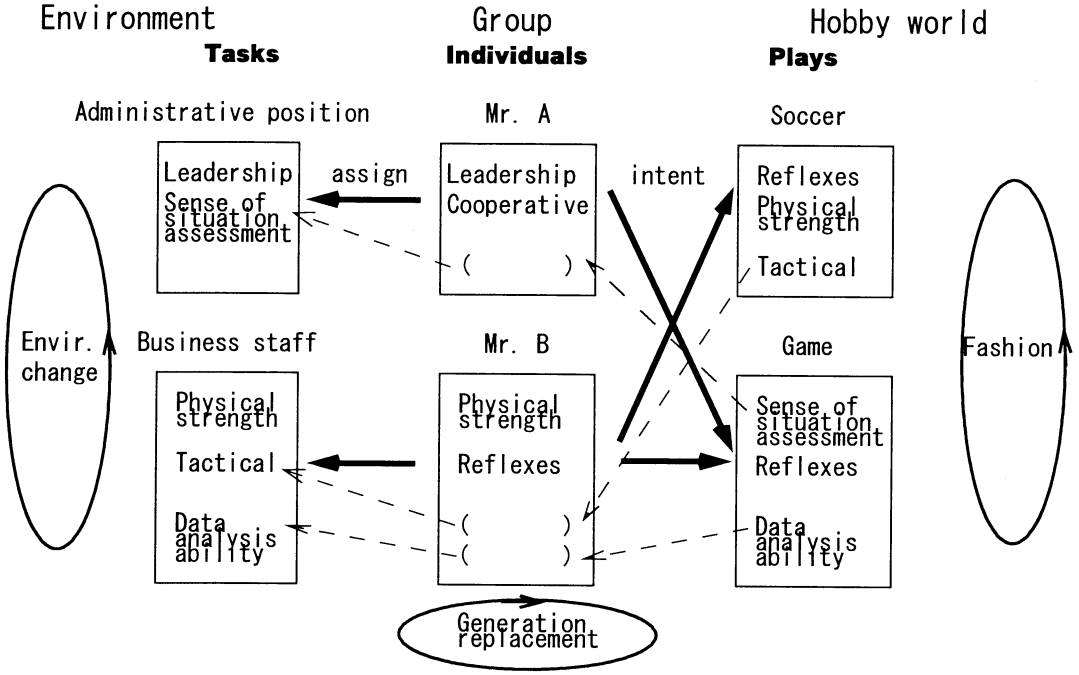
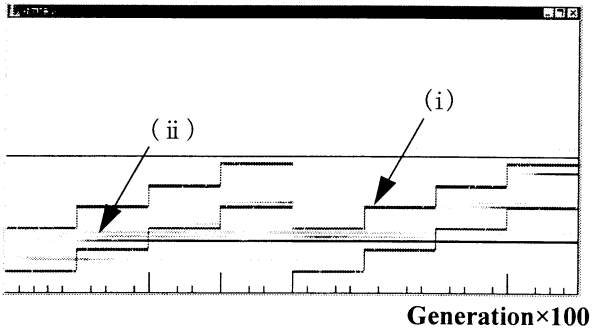
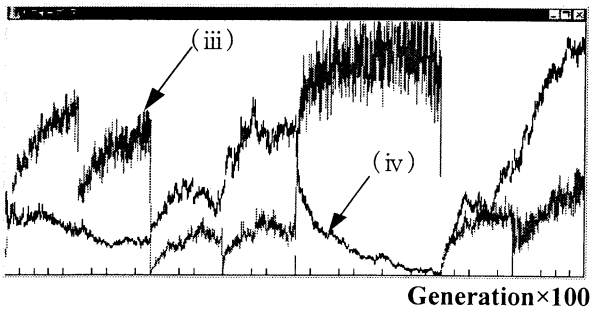


Fig.1 World model with interaction to external environment.

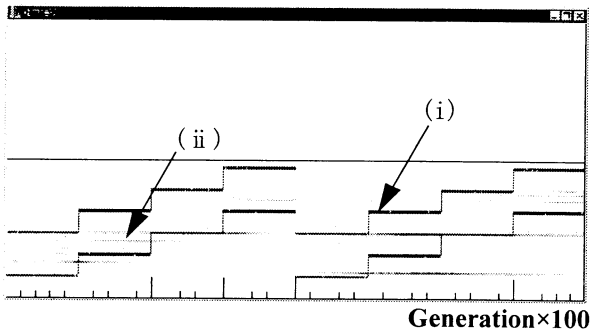


(a) (i)Requested abilities and
(ii) generated ability spectrum

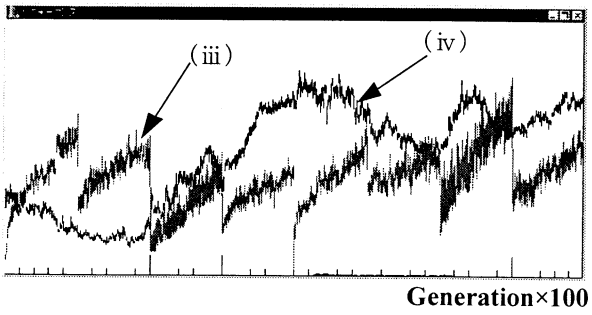


(b) (iii) Performance and
(iv) variance of ability distribution

Fig.2 Behavior of the world with 64 kinds of less complicated plays each of which requires no ability.



(a) (i)Requested abilities and
(ii) generated ability spectrum



(b) (iii) Performance and
(iv) variance of ability distribution

Fig.3 Behavior of the world with 64 kinds of highly complicated plays each of which requires 64 abilities.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

The study of the language occurrence model using Associatron

Masanori Sugisaka and Hisashi Aito
Department of Electrical and Electronic Engineering, Oita University
700 Dannoharu, Oita 870-1192,JAPAN
msugi@cc.oita-u.ac.jp,haito@cc.oita-u.ac.jp

Abstract

The purpose of this study is to consider the mechanism of the language occurrence which simulates relation between the language and the brain of the human being who controls a language on computer.

Therefore, the process of storing things by two intelligent robots and a phenomenon in the outside world were modeled using Associatron, which is a kind of the associative memory model of the neural network, by forming a self in the brain and by creating a common word. The simulation study is performed.

1. Introduction

For the human being, the language is very interested. Because the language is the means for showing a human mind activity, the human being built up an advanced culture by the language which included a letter language. The influential way of clearing up this language includes an approach used in biology and medicine for the brain. However, it is very difficult to observe a language by the cerebral level. Therefore, we simulates it on computer and understand the mechanism of the language occurrence. We use Associatron which is a kind of the association memory model of the neural network which obtain a hint to the mechanism of the brain.

2. The principle of Associatron

Associatron is the association memory model which was proposed by Nakano.[2]. The Associatron didn't merely realize operation of the association. The characteristic is the fact to have modeled a cerebral nervous circuit. First, we show the group of the neuron in

Fig.1. x_i is the condition of the cell C_i, $\mathcal{M}_{ij}$ is the strength of the combination between C_i and C_j ($\mathcal{M}_{ij} = \mathcal{M}_{ji}$)

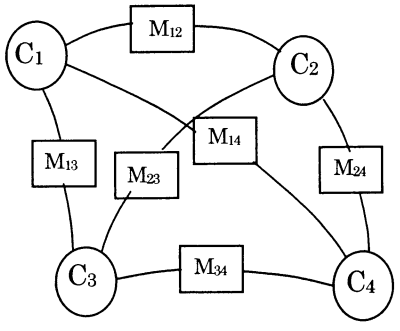


Fig.1 The modeling of the group of the nervous cell

The pattern memorized is shown in vertical vector

$$x_i = (x_1, x_2, \cdots, x_n)^T \qquad (1)$$

where x_i takes 1, 0, -1. In this way, pattern space is coded by expressing one memory pattern in one vector. Suppose that there are n memory patterns. We take the self correlation of this each of the n patterns. Then Eq.(2) is obtained when adding from 1 to n.

$$M = \sum_{k=1}^{n} x_k x_k^T \qquad (2)$$

where x_k shows one memory pattern. n patterns are preserved in this procession $\mathcal{M}$ and $\mathcal{M}$ is called a memory procession.

A recall is done as follows. It defines a quantum function ϕ as

$$\phi(x) = \begin{cases} -1 & (x < 0) \\ 0 & (x = 0) \\ 1 & (x > 0) \end{cases} \qquad (3)$$

where ϕ is the function of vector $x = (x_i)$. We consider a quantum of every element with respect to procession $\mathcal{A} = a_{ij}$.

$$\phi(x) = (\phi(x_i))$$

$$\phi(A) = (\phi(a_{ij})) \qquad (4)$$

Suppose that the recall input is $y = (y_1, y_2, \cdots, y_n)$, the recall by memory procession $\mathcal{M}$ and input y becomes

$$z = \phi\{\phi(M)y\} \qquad (5)$$

where z is n dimension vector, too.

If we suppose that Eq.(5) includes the part of some memory pattern x in input y, x can be correctly recalled by Eq.(5).

Next, the modeling of concept forming and language occurrence using this Associatron is explained.

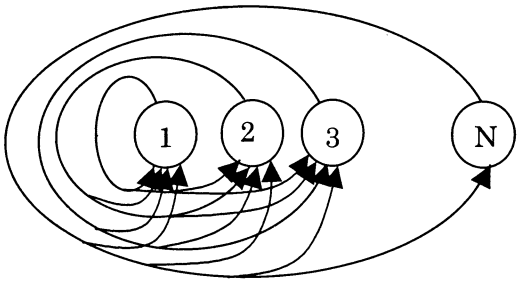
3. The modeling of concept forming and language occurrence

When one robot sees that the other robot is struck by "lion", the various information obtained by the robot is taken in the brain as the excitement pattern of the neuron. The concepts such as the attribute which "lion" has, i.e. "dangerous", "brown", and "the hair has grown", etc., we think that a concept "lion" was recorded to the brain. The robot stores various information as the concept. In this way, to model the function of the brain by forming "the world image" of the outside world like an organization of the self is one purpose of simulation. On the other hand, the robot gives a name (the language) to each concept obtained selfishly. But, at first this name is different from the name given by a stranger based on the same concept. By repeating "the common experience" which change both the name from hearing the word each other and the attribute by seeing a lion twice at the same time, the words of both agent agree gradually. To model the function of the brain in order to make a concept and a corresponding word is one problem changing the word adaptively is another problem.

Therefore, we describe how to model such a system. First, concepts such as "brown" and "lion" are expressed as the excitement pattern of the neuron. Here, we express various things and the various attributes in the stimulation pattern of 55 bits. It corresponds to the stimulation pattern number of the bits and a cerebral model is composed by 55 cells. Then, as the memory model, we use Associatron and allocate another Associatron to make a name (the language) correspond to each concept, too. Each Associatron is called Associatron1, Associatron 2.

4. The model of the concept forming

Associatron1 has the structure like Fig.2. A stimulation pattern is shown in the bit pattern of 1, 0, -1.



The feedback combination

Fig.2 Associatron 1

For example, in the robot, when seeing a lion, the stimulation pattern which corresponds to the lion is memorized in Associatron 1. Supposing that the stimulation pattern is $u=(u_1,u_2,\ldots,u_{\mathcal{N}})^T$ ($\mathcal{N}=55$), the load processing of Associatron 1 is $\mathcal{M}_w$, is expressed by Eq.(6).

$$M_w \leftarrow M_w + uu^T \tag{6}$$

In this way, each concept is memorized in Associatron 1 latently. The Associatron 1 attempts stimulate the optional cell which memorized various concepts. That is, it gives recall input like Eq.(7). The Associatron 1 provides a suitable boundary and recalls a pattern.

$$s = (0,0,\cdots,1,0,\cdots,0)^T$$

$$s = (0,0,\cdots,-1,0,\cdots 0)^T \tag{7}$$

This is called optional stimulation. Now, the cell which is combining with the stimulated cell strongly fires and the pattern of some concept is taken out.

Therefore, the concept can be extracted from Associatron by the optional stimulation and simulation memorizes an extracted pattern in the memory area which is called a world image memory part. The change pattern is the 10 bit pattern to input when recalling a word in Associatron 2.

5. The model of the language occurrence

The correspondence between each concept and a word is modeled in Associatron 2. The structure is shown in Fig.3. A word is shown in the 5 bit pattern which consists of 1, -1, 2 values.

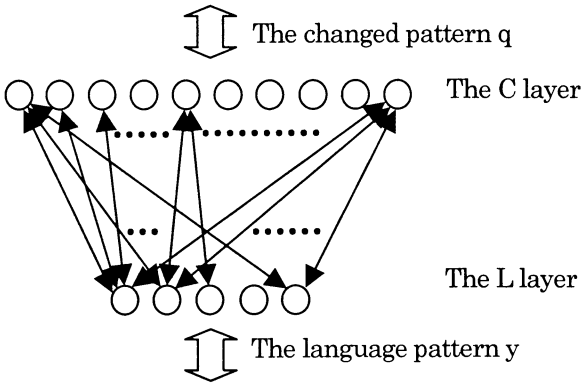


Fig.3 Associatron 2

The way of recalling the word which corresponds to the concept is showing in Fig.3. The changed pattern showing a concept is given the C layer in Fig.3. The pattern which shows the word is obtained as the output. From the L layer supposing that the combination load processing of Associatron 2 is $\mathcal{M}_l$, the language pattern is y and the changed pattern is q. A recall formula is given by Eq.(8).

$$y = \phi(M_l q) \tag{8}$$

But, ϕ is the function such that

$$\phi(x) = \begin{cases} 1 & (x > 0) \\ -1 & (x \le 0) \end{cases} \tag{9}$$

A robot works in this way and it recalls a word from the concept and generates a new word. This is the mechanism of the common experience. Associatron 2 should memorize correlation between the concept recalled, and the word from the companion at the

time . A figure of this whole system, is shown in Fig.4.

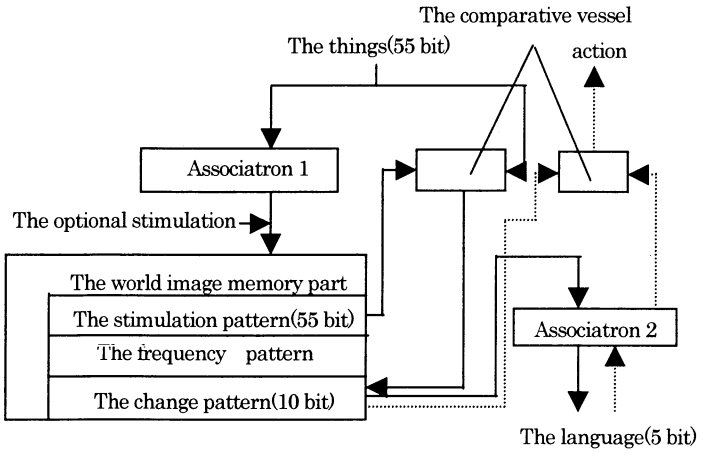


Fig.4 The cerebral structure of information processing

6. Simulation Results

First, it does the explanation of the simulation is simply given. In the outside world six things are given by "avalanche", "lion", "bear", "rabbit", "nut", and "black pig". These things are defined in some combinations in six attributes of "dangerous", "delicious", "hair", "white", "black", and "brown". A word consists of 1 or −1 for a pattern. The word 1 corresponds to the sound, "BA", and the word −1 corresponds to the sound, "BU". And then, the combined word 1 −1 1 −1 generates "BABUBABU".

The result of the simulation is described below. As a result of simulation, we could observe the process of the agreement percentage of the language changing by the number of times of the common experience of two robots and the appearance frequency of each attribute. The memory of Associatron maintained the information which can form the world image and could analyze it and also showed what doing learning with being able about for the language to agree. It was possible to associate with the each attribute of the things right by.

7. Conclusions

In this paper, using this Asociatron, simulation studies for the process of the occurrence of the language of the human being are performed on the computer. The result of this study could have simulated the language occurrence model of the human being.

In the near future, and we want to make a language occurrence model be more complete.

References

[1] M.Sugisaka, " The system theory and the control " , in Japanese , Nikkankougyou Shinbunsya, pp.157~pp.162 ,1997

[2] K.Nakano, "The information system of the brain to make with C " , in Japanese , KindaiKagakusya Co.,Ltd ,pp.76~pp.79, pp.221~pp.232,1995

[3] K.hisama, K.nakayama, "The neuro computer engineering" , in Japanese, Kogyochosakai Publishing Co., Ltd., pp.15~pp.36, 1992

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Artificial Knowledge system constructed Interconnectional Network modeled on the natural Brain

Naoki Nishinomiya and Yasuo Yonezawa*
Complex Adaptive system Laboratory, Division of System Engineering,
Graduate school of Science and Engineering, Ibaraki University,
4-12-1, Nakanarusawa, Hitachi, 316-8511, Japan.

Keywords: Reversible connection network, Pattern recognition, Artificial brain.

Abstract

In this paper, the artificial brain [1] [2] model based on interconnectional network with reversible potential of neuron by impulse response is described. For presented model is referred the structure and function of natural brain, we designed a interconnected neuron network model (two or 3rd dimension) that consists of nonlinear oscillators obtaining different frequencies and that can produce stable activation patterns by variable interactions derived their frequencies and voltages by impulse responses. Also, our presented model is tested with pattern recognition of several symbols and words by two dimensional model. Simulations of this model (in case of two dimension) are indicated the good result with both function of associate memory and memory recall at similar and modified symbols and words. These tests are carried out the pattern matching by comparing between the matching of activation pattern in case of learning and judgement. The leaning of model are given to produce distinctive activation pattern of network pathway that correspond to each learning symbols and words. Memory recall tests [8] are performed with comparison (pattern matching) of activation pattern generated by unknown data, learning modified data and learning data.

1 Introduction

Recently the term neural networks has different meanings for several researcher [3]. For the computer scientist, control engineer or systems engineer [4], it means a network of simple processors showing some of the characteristics of neurones in real organisms that is important because of its special computing capabilities. Usually, at one end of the network, data is fed from measuring instruments in the external world, and at the other end the network produces output which might be read directly by a series of layers of 'neurons' which fire or not depending on their current state, their input from earlier layers and their own properties. Those that fire send impulses to connected neurons in subsequent layers [5]. The networks are not programmed in the way that other computers are, by analyzing the task required and giving the computer specific instruction to perform it. Surprisingly, they can be trained to perform complex task by random adjustment of the connection strengths between neurons [8]. Those patterns of connections which produce a better approximation to the desired relationships between input and output are reinforced; those which do not are discourage by weakening the connections. We are very interested in networks of neuron in real organisms, exhibiting a wide range of properties and subject to a plethora of influence. Real neurons have a cell body, to which are attached dendrites with received input from the terminals of other neurons.

From the cell body projects the axon, which is very long in some cases, and, at the end of the axon, the cell branches out to a number of terminals, which then established contact with, and send impulses on to, the dendrites of other neurones. The interior of the resting cell is at different potential from the outside fluid, and consequently there is a potential difference across the cell membrane, called the membrane potential.

Any input via its dendrites in the form of pulses of increased voltage are quickly transmitted by the cell along the dendrites to the cell body. When these integrate to a value at the cell body which is greater than the cell threshold, the cell fires: for a very brief time, the membrane potential reverses its sign, after which the cell reverts back to its previous equilibrium. But before the cell body returns to equilibrium, an

*To whom correspondence should be address; E-mail yonezawa@life01.dse.ibaraki.ac.jp

impulse, in the form of the brief potential reversal, is transmitted along the axon to terminals of the cell, which can then be transmitted to other cells further on in the network [9]. The point at which a terminal of one cell lies adjacent to a dendrite of another and hence the route by which impulses are transmitted from cell to cell is called a synapse [6].

Moreover the new finding which conflicted with usually assumption of the modelers was that neurons are by no means all the same. For example, in the ganglia of Aplysia, neurons very in many ways: size, shape, pigmentation, the neurotransmitters they utilize, and their firing patterns. Using these differences, researchers have been able to identify the same series of neurons in many different individuals, thus showing a partial invariance from individual to individual in the cells found in particular positions. Also, some parts of the vertebrate brain also contain quite heterogeneous collections of cells serving variety of functions. Communication in some areas of the brain is not so specific as neuron-neuron contact, nor is it so fast: some neuropeptides diffuse over lager area on longer time scales, and appear to function more as broadcasters, possibly bring groups of neurons into synchrony [7]. From the standpoint of real organism, it is clear that the description as above is helpful when constructing of Artificial brain. If we take these additional informations in brain science field, previous brain models must be modified. In order to elucidate the information processing of flexible function on brain, we challenge the new model based on the new additional information in brain science. In this paper, we proposed the construction of artificial brain similar at natural brain information's.

2 Materials and methods

2.1 Information processing in natural brain

In generally, it is known that the natural brain is composed of numerous interconnected neurons performing various nonlinear transformations, it is formed a 3rd dimensional layer structure by the neural networks based on parallel distributed processing. Each neuron mutually communicates information as electric impulses through synaptic transformations, the temporal oscillations of neuronal activities, sequential electric impulses, represent information. In the brain, we assume that external information is represented particular pattern of activity of neurons by their interactions as internal information, these patterns (internal information) correspond to external information. Thus these interactions, neuronal activities, are preserving the pattern and it is possible to learn as a result of tuning these interactions by learning data. It is important that each neuron has local interactions based on neuronal oscillations and synaptic transformations, the global activity of these interactions in the neural network can generate stable activation pattern depended on leaning data. Thus the pattern recognition seems to compare distinctions of activation patterns, we consider the comparison (pattern matching process) to be the basis of intelligence information processing. At simply, the neural network can generate various activation patterns depended on leaning data using abilities of information transformation of neurons, not leaning data and modified known data are recognized a certain information by pattern matching with memorized patterns.

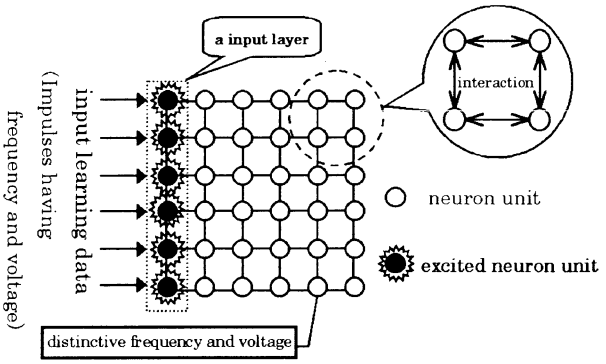


Figure 1: Outline of Artificial Knowledge system. Each neuron unit is exchanged the frequency and voltage by neuron interactive action in according to leaning data.

2.2 Model of Artificial Knowledge system

For viewpoints mentioned above, we constructed an interconnected neural network model as an artificial knowledge system. This model can produce particular pattern of activation of network pathway from leaning data by their interactions. Because the leaning data is expressed activation pattern in the network as internal information, it is possible to do pattern recognition based on pattern matching process which compares among activation patterns depended on each leaning data.

In our model, we adopted to the basic functional

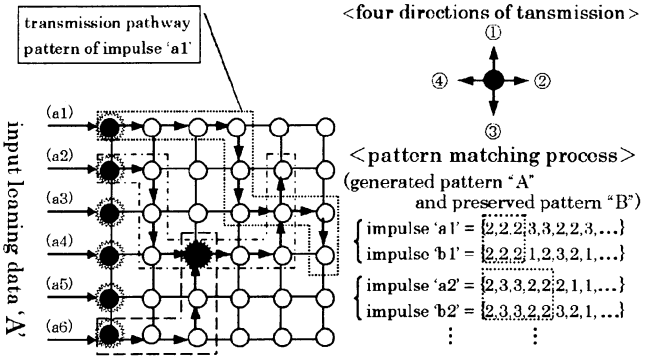


Figure 2: Patterns of transmission pathway of impulses as an output, example of comparison of activation pattern generated by leaning data "A" and preserved pattern "B".

Table 1: Range of output voltages of impulse depended on excited unit's frequency

excited neuron unit's frequency	decoded voltage of impulse
15.0 ~ 24.0	5.0 ~ 20.0
7.0 ~ 15.0	20.0 ~ 50.0
4.0 ~ 7.0	50.0 ~ 70.0
0.3 ~ 4.0	35.0 ~ 65.0

unit (as a neuron) material property of frequency that is varied by concurrent transmissions of voltage for the sake of giving property of information processing between real neuron and synapse. We considered plural neuron having similar property to be singular unit having distinctive frequency in order to give a characteristic behavior to each unit. These units are arranged such as that shown in Fig.1, each unit has interconnections with near (4 neighborhood in matrix square) units and the network have one input layer. The network structure of natural brain is observed as heterogeneous network preparing by bifurcation of synapse linkage with neuron growth. Also, each neuron cell has been unique character (ex: individual frequency and electric potential). Namely, these character of neuron-synapse network are generated the nonlinear properties of impulse transmission. Our model are represented with above mentioned network structure as same as typical behavior of impulse transmission of natural brain. These network properties are both character of impulse transmission depended on unique frequency and electric potential. Therefore, our model can generate the dynamics of impulse transmission into natural brain by lattice square structure. The external information, symbols and words, are transformed into impulses having frequency and voltage as a leaning data, when input impulses are communicated to each unit in input layer, these units are excited. Because an unit has property of oscillator, an excited unit transmits impulse near an unit whose frequency is closest to frequency of the impulse. Thus an unit has connections derived from frequencies of neighboring units, it has the switching function depended on each frequency of impulse in the network.

Furthermore, the excited unit makes voltage of impulse pushed up specific range of value according to the unit's frequency (Table1). Especially, in case some impulses are simultaneously transmitted to same unit, the frequency of unit is changed. This change is driven by

$$\ddot{x} + k\dot{x} + C_1 x + C_2 x^3 = B\cos\nu t,$$

and we assume following to be result,

$$x(t) = a\sin\nu t + b\cos\nu t,$$

where C_1(=40.0), C_2(=0.001) and k(=1.0) are the constant, B is the transmitted voltage, a and b are the excited unit's voltage amplitudes and ν is the excited unit's frequency. In the above equations, we calculate the excited unit's frequency (ν). These changes generate variable activation patterns. After all, a leaning data is represented as impulses, these impulses communicate information according to these frequencies in the network, and the pattern of transmission pathway of impulses is produced by interaction of entire units and a leaning data. Thus, we obtain the pattern of transmission pathway of each impulses, described the combination of four vectors in time series, as an output (Fig2). When we calculate the values of similarity among activation patterns, first we compare each impulse transmitted to same input point about two activation patterns, and then we appreciate two transmission pathway patterns of impulses in time series until matching two vectors are different. And we carry out the matching process for all impulses of each activation pattern. From the above way, we calculate the values of similarity among activation patterns with pattern matching.

2.3 Simulation procedure

We set basic functional unit of neuron on 100x100-lattice square and gave distinctive frequency to each unit at random (1.0~24.0). We used alphabets exchanged at bit map ("0" and "1" pattern) on 100x100-lattice square as a leaning data. Because a leaning

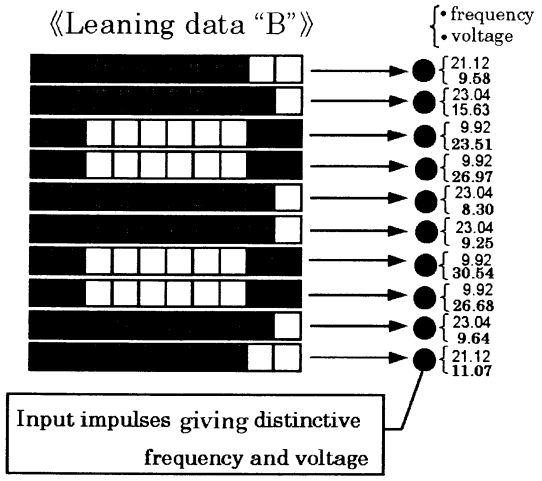


Figure 3: Example for a bit map pattern on 10x10-lattice square of a leaning data "B" and the altered impulses pattern. The black cell represents "1", and the white is "0".

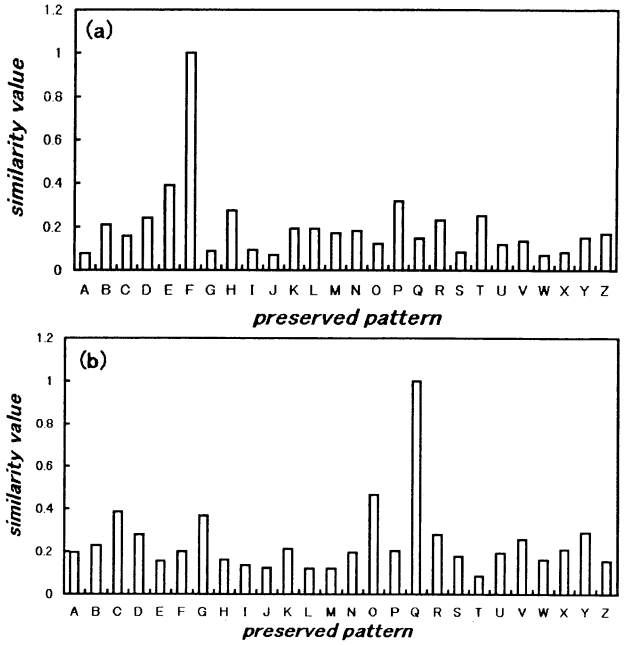


Figure 4: Alphabets recognition; (a) leaning data is "F", (b) leaning data is "Q".

data is communicated to the network as impulses pattern, the pattern of leaning data is altered to impulses (Fig.3). In order to give impulses to features of leaning data, we transform a row of the bit map pattern of leaning data into one impulse having frequency and voltage according to number of '1' and the each position. Thus one leaning data (bit map data) is modified to one hundred impulses in this model. We used alphabets and some words as leaning data and took out the global pattern of transmission pathway of impulses as output. First, we communicate each leaning data to the network and each basic generated pattern is preserved. Next, some leaning data is communicated to the network, the leaning data can be classified into some categories to compute values of similarity with pattern matching between preserved patterns and generated pattern. Thus levels of pattern recognition depend on values of similarity. If our model accepts a lower level of values of similarity as same category, its pattern recognition becomes coarse discriminations and it becomes fine discriminations at a higher level of values of similarity. In the words recognition, each alphabet of the word is successively communicated to the network at fixed intervals. Thus we used all alphabets as basic leaning data, and moreover, in order to confirm that our model can do pattern recognition for the separation of different leaning data and the unification of similar leaning data, we used all alphabets including noise as leaning data.

Table 2: Accuracy of alphabets recognition.

classified rate of similarity	original data	+5% noise	+10% noise	+30% noise
1.0	1.0	0.96	0.88	0.53
0.8	1.0	0.88	0.65	0.19
0.6	1.0	0.46	0.26	0.0
0.4	0.69	0.03	0.0	0.0

3 Simulation result

In these simulation results, we observed that the leaning data was roughly classified into correct category at using fine discriminations, the values of similarity tend to be higher among similar leaning data. Part of the results indicated in Fig.4 and Fig.5. We used all alphabets, some words and alphabets including noise as a leaning data. In order to examine a ratio of pattern recognition, we changed a value of classification criterion (classified rate of similarity) each leaning data. The results are shown in Table2 and Table3. In these results, our model could properly recognize with leaning data even if a leaning data somewhat has noise. However, in the pattern recognition of alphabets including much noise, our model could not achieve good results. In this case, we supposed that the features of alphabets were disappeared by noise and the value of similarity of an only correct category decreased. And

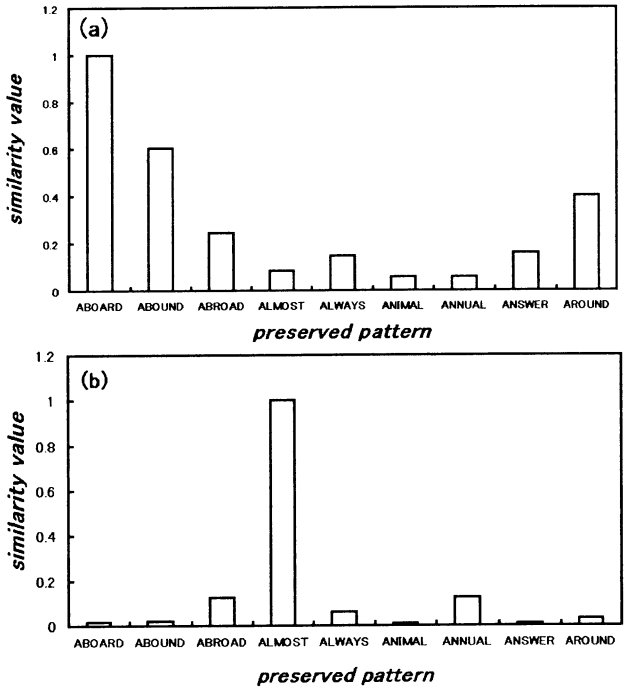


Figure 5: Words recognition; (a) leaning data is "ABOARD", (b) leaning data is "ALMOST".

Table 3: Accuracy of words recognition.

classified rate of similarity	3 letters word	6 letters word
1.0	1.0	1.0
0.8	0.88	0.96
0.6	0.72	0.88
0.4	0.61	0.59

in the pattern recognition of words, it is necessary that we make criterion of pattern recognition strict in order to classify a leaning data into correct category. For example, in Fig.5 (a), the values of similarity of "ABOUND" and "AROUND" are higher than other words. However, a tendency to be higher value among similar leaning data not only has a bad aspect but also has a good aspect. That is to say, after adjusting a value of classification criterion, it is possible to separate automatically the similar leaning data into one category.

4 Discussion and Conclusion

In previous neural network model based on the hierarchic structure, each basic unit transmits input information altered by connection weights to other units and the information are communicated to only decided directions. And the last condition of network points an output, that is to say the changing conditions of neuron units, information flow, are not so important toward output and the output is restricted to the network size. Thus a capacity of the network fundamentally depends on a scale of the network size. However, our model can generates particular activation pattern depended on leaning data, the changes from an initial condition of the network, described by transmission pathway of time series of general impulse, shows an output. The most spectacular plasticity in the nervous system occurs during its development. The learning which so fascinates and mystifies us when we contemplate acquisition of general and episodic knowledge, motor and cognitive skills, language and reading, plues by comparison with the electrifying conundrum that is development [10] [11]. Hundreds of different kinds of cells must the produced-Purkinje cells, motor neurons, amacrine cells, chandelier cells, and etc. with the morphology and physiology characteristic of their class. From these properties, the capacity of our model doesn't depend on a scale of the network size, because it is possible to represent a good deal various transmission pathways in the network [12] [13]. Thus the model can preserve many memories without disturbing encoding memories, it can recall particular memory correctly.

In our model, first a leaning data is altered to impulses having distinctive frequency and voltage, and then these impulses are communicated to the network. Because transmission direction of the impulse is decided by neighboring units' frequencies, the transmission pathway of time series of an impulse depends on all neuron units' frequencies in the network. And as a result of frequency changed by impulses collision, the condition of network (frequencies of units) is changed every time step by impulses transmission. This means that our model varies the dynamic connections of neuron units in order to adapt input impulses. That is to say, individual impulse moves around units according to the property, some impulses change unit's frequency by impulse collision. And the general impulse activity produces characteristic time series pattern of network pathway which corresponds to leaning data because the impulse activities distinct on features of each leaning data.

From these properties of our model, we consider that our model carries out self-organization, that is to say the model has the dynamic condition of network derived from variable frequency property of neuron units, the condition is changed because of impulses interaction, with the result that the model gen-

erates characteristic activation pattern which reflects features of leaning data. Thus it is possible to keep correspondence between leaning data and activation pattern because of the self-organization of network.

5 Summary

In this paper, we are addressing the question, which is difficult about the generative intelligence like natural brain in computation. We presented the artificial knowledge system constructed interconnectional network based on the properties of structure and action as natural brain. This model obtained the reversible neuron transitions that collide with impulse transmission. The result own computer experiments clearly show that the creative intelligence largely depends on the transmission of neuron activity (frequency and voltage as electric potential). A key issue leading to network models concerns how the behavior of individual adapts successfully in real time to constraints imposed by their environments. In order to analysis this essence, one needs to identify the functional level on which an individual's behavioral success is defined. Much theoretical and experimental result's suggests that this is the level of neural networks, rather than the revel of individual never cells. Key behavioral properties are often emergent properties due to interactions among many cells in neural network at three-dimensional structure. We would now like to go on to develop this model by extending own research to three-dimensional structure in the artificial brain modeled reversible connectional network same as natural brain.

Acknowledgements

We acknowledge support from grant of SVBL (Complex Functional Robots Laboratories) in Ibaraki University. We thank Professor Y.Tsutsumi and Professor M.Agu for the support of research activity in our university. Also, we thanks research stuffs for paper preparation in our laboratory.

References

[1] Stephen Grossberg, *Neural Networks and Natural Intelligence*, A Bradford Book The MIT Press, 1988.

[2] Patricia, S. Churchland and Terrence, J. Sejnowski, *The Computational Brain*, A Bradford Book The MIT Press London, 1993.

[3] Kandel, E., *Small Systems of Neurons, From Neurons to Networks*, Scientific Americas, pp. 70-86, 1989.

[4] Kandel, E. R., Schewartz, J. H. and Jessell, T. M., *Principalls of Neural Science*, Elsevior. N. Y, 1991.

[5] Amari, S., "Neural theory of Association and Concept-formation," *Biological Cybernetics*, Vol. 26, pp. 175-185, 1977.

[6] Caiaviello, E., "Outline of a theory of though-process and thinking machine", *J. theoretical. Bio.*, Vol. 2, pp. 204-235, 1961.

[7] Kanter, T. and Somopolinsky, H., "Associative recall of Memory without errors", *Phys. Rev. A.*, Vol. 35, pp. 308-392, 1987.

[8] Siiino, M. and Fulcui, T., "Self-consistent signal to noise analysis of the statistical behavior of analog neural networks and enhancement of storage capacity", *Physical Rev. E.*, Vol. 48, pp. 687-697, 1993.

[9] Yoshisawa, S., Morita, M. and Amari, S., "Capacity of associate memory using a non-monotonic neuron model", *Neural Networks*, Vol. 6, pp. 167-176, 1993.

[10] Harris, W. A., *Neurogenesis and determination in the Xenopus retina*, In Lam and Shatz, pp. 95-105, 1991.

[11] Reh, T. A., *Determination of cell fate during retinal histgenesis: intrinsic and extrinsic mechanisms*, In Lam and Shatz, pp. 79-94, 1991.

[12] Damasio, A., "Time-locked multiregional retroactivation: A system level proposal for the neural substrate recall and recognition", *Cognition*, Vol. 33, pp. 25-62, 1986.

[13] Damasio, A. and Damasio, H., "Brain and language", *Scientific American*, Vol. 267, pp. 88-95, 1992.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Multi-module reinforcement learning with situation decomposition algorithm

○ Osamu ITOU, Tokyo University of Agriculture and Technology
Hiroyuki OKADA, FUJITSU LABORATORIES LTD.,
Tokyo University of Agriculture and Technology
Hiroshi YAMAKAWA, FUJITSU LABORATORIES LTD.
Takashi OMORI, Tokyo University of Agriculture and Technology

Abstract

We propose a learning system design concept that decomposes environmental situations from many sensory data before learning, prepares modules that receive only meaningful sensor data for each situation, and automatically selects and learn the modules. In this paper, we realized the situation decomposition by estimating "Matchability" criterion, learning of modules by actor-critic model of reinforcement learning method, and applied it to a navigation problem of mobile robot.

1 Introduction

In mobile robot study, reinforcement learning is often used in various tasks as a method to acquire behaving strategy from trial and error manner. However, the learned strategy is not so robust for environmental change, and learning speed is slow when the task has many input, hence the learning proceeds in huge search space.

To solve the problem, we focused on human learning strategy that selects information source depending on situation, and learns reflection type action from selected experience that matches to the situation. A merit of the situation dependent input selection and experience selection is the quick learning that is based on simple structure of data space and small number of learning data. The function is also effective to avoid unnecessary input noise that is caused by unrelated input information dimension for a robot behavior on the moment. To realize the quick learning on mobile robot, we need mechanism to extract the situation that is composed of information sources and experience cases, and to select information source for learning depending on the situation. We call the problem as situation decomposition for learning machine.

For the situation decomposition problem, multi-module rainforcement learning is often used[1],[2]. In the model, each module is often designed by human who knows target environment in advance. However, designing of learning module is almost same as identification of target machine and its environment. It is not easy for human to understand whole the system.

In this paper, we propose an algorithm of automatic environmental situation decomposition, a module design algorithm for each of situations, and a module selection method to realize effective and robust learning. We also show computer simulation result to evaluate effectiveness of the system.

2 Situation decomposition with matchability criterion

2.1 Situation decomposition

The proposed system automatically generates modules using situation decomposition method.[4] The method decomposes spread-sheet type input data into multiple meaningful partial data sets. A spread-sheet type data is a set of cases, each of which is vector of values for each feature. Output of this decomposition algorithm is sets of partial data that fit to "matchable situation". Each of the extracted situations that are corresponding actor-critic module learns suitable action and expected reward.

We define a features selecting vector $\mathbf{d}$ and a case selecting vector $\mathbf{n}$ for each situation g. Then each of extracted situation g is indicated by pair of $\mathbf{d}$ and $\mathbf{n}$. This decomposition method selects local maximums of matchability criterion (see next subsection) from huge 2^{D+N} combination of partial situation. Outline of the decomposition method is described in Figure 1(a).

2.2 Matchability criterion

Matchability criterion is used for situation decomposition method. This criterion estimates compactness of each of partial situation. The criterion value increases when the feature selection and case selection increases data explanation ability.

Matchability criterion contains a number of selecting cases $|\mathbf{n}|$ and a number of selecting features $|\mathbf{d}|$ to represent "coverage of data". And it also contains a number of selecting segment $|\mathbf{r_d}|$ (see after) to represent "simplicity of model". Using these parameters the situation decomposition algorithm extracts matchable situation $g^* = (\mathbf{d}^*, \mathbf{n}^*)$ each of which contains rules.

For calculation of the criterion, we introduce segment space, in which each feature dimension is divided into u_i segments, and let the vector of the segment numbers is $\mathbf{u} = (u_1, u_2, \cdots, u_{|\mathbf{D}|})$. Here, $\mathbf{D}$ is all features selecting vector, and $|\mathbf{D}|$ is a number of all features. The partial feature space is indicated by the feature selecting vector $\mathbf{d}$.

The segment selecting vector in partial feature space is $\mathbf{r_d} = (r_{\mathbf{d}_1}, r_{\mathbf{d}_2}, \cdots, r_{\mathbf{d}|\mathbf{U_d}|})$. Here, $r_{\mathbf{d}_i}$ is the binary value indicates selection/unselection of each segment. Here, $\mathbf{U_d}$ is segment selecting vector which selects all the segments in patial feature space and $|\mathbf{U_d}|$ is a number of all segments of in partial feature space. The number of the selecting segments is $|\mathbf{r_d}| (0 < |\mathbf{r_d}| \leqq |\mathbf{U_d}|)$.

Originally the search process in partial feature space is the searching of the local maximization of Matchability criterion in case selection space. Algorithm select the segments instead of cases, under the condition that all the cases that are included in the segment are selected. This trick dramatically reduces the search space.

For concrete implementation, we replace a number of selected features $|\mathbf{d}|$ by the number of selected segments $|\mathbf{U_d}|$ in partial feature space. The matchability criterion is defined as follows by combining these three variables.

$$M(|\mathbf{n}|, |\mathbf{r_d}|, |\mathbf{U_d}|) = C_1 \log \frac{|\mathbf{n}|}{|\mathbf{N}|} + C_2 \log \frac{|\mathbf{n}|}{|\mathbf{r_d}|} - C_3 \log \frac{|\mathbf{r_d}|}{|\mathbf{U_d}|} \quad (1)$$

C_1, C_2, C_3 is positive constant. $|\mathbf{N}|$ is a number of all cases. $|\mathbf{n}|/|\mathbf{N}|$:Proportion of cases that are included in the partial situation. $|\mathbf{n}|/|\mathbf{r_d}|$:Mean value of number of selected cases in each segment. $|\mathbf{r_d}|/|\mathbf{U_d}|$:Occupation ratio of the segment in partial feature space.

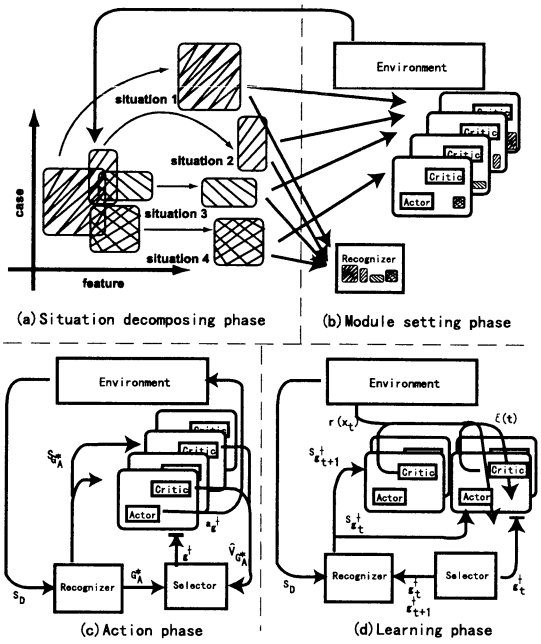


Figure 1: Proposed model

3 Multi-module reinforcement learning

3.1 Basic behaviar of proposed system

The proposed system behaves as follows.

1. In situation decomposing phase (Figure 1(a)) and module setting phase(Figure 1(b)), a module which is conposed of an actor and a critic are made for each situation. At the same time, recognizer is made as database of all situations.

2. In action phase(Figure 1(c)), recognizer outputs situations $\mathbf{G^*_A} = (g^*_{\mathbf{A}_1}, g^*_{\mathbf{A}_2}, \cdots, g^*_{\mathbf{A}|\mathbf{A}|})$ that match with inputted case from all matchable situations. Selector selects a situation $g^\dagger_t$ to maximize expected reward from outputted situations. An actor of module m_t for selected situation outputs an action. A robot moves to new state according to the action.

3. In learning phase (Figure 1(d)), a critic of module m_t is given input state $\mathbf{s}(\mathbf{x}_t)$ for position vector $\mathbf{x}_t$ of the environment from recognizer, receives reward $r(\mathbf{x})$ on position $\mathbf{x}_t$ from the environment, receives a summation $\hat{V}$ of estimate rewards from critic of module m_{t+1} that is selected at time $t+1$,

and learns to minimize estimated error $\epsilon(t)$. The actor of module m_t learns strategy of actions.

4. After that, alternately repeating action phase and learning phase, the robot moves and learns to achieve a task.

3.2 Learning of critics

A state evaluation is calclated in module m_t and m_{t+1}. Here, we defined a summation $\hat{V}_{g_0^\dagger}(\mathbf{s}_{g_0^\dagger}(\mathbf{x}_t))$ and $\hat{V}_{g_1^\dagger}(\mathbf{s}_{g_1^\dagger}(\mathbf{x}_{t+1}))$ of estimate rewards at time t and $t+1$ by following equations.

$$\hat{V}_{g_0^\dagger}(\mathbf{s}_{g_0^\dagger}(\mathbf{x}_0)) = r(\mathbf{x}_0) + \gamma r(\mathbf{x}_1) + \gamma r^2(\mathbf{x}_2) \cdots \quad (2)$$

$$\hat{V}_{g_1^\dagger}(\mathbf{s}_{g_1^\dagger}(\mathbf{x}_1)) = r(\mathbf{x}_1) + \gamma r(\mathbf{x}_2) + \gamma r^2(\mathbf{x}_3) \cdots \quad (3)$$

Here, $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2$ and $\mathbf{x}_3$ is position vectors at time t, t+1, t+2 and t+3. γ is a time discount factor.

From equations above, we got temporal equations of expected rewards and estimation error $\epsilon(t)$.

$$\hat{V}_{g_0^\dagger}(\mathbf{s}_{g_0^\dagger}(\mathbf{x}_0)) = \gamma \hat{V}_{g_1^\dagger}(\mathbf{s}_{g_1^\dagger}) + r(\mathbf{x}_0) \quad (4)$$

$$\epsilon(t) = \gamma \hat{V}_{g_1^\dagger}(\mathbf{s}_{g_1^\dagger}(\mathbf{x}_1)) + r(\mathbf{x}_0) - \hat{V}_{J_0^\dagger}(\mathbf{s}_{g_0^\dagger}(\mathbf{x}_0)) \quad (5)$$

The critic of module m_t learns to minimize the estimation error $\epsilon(t)$, provided the following conditions.

$$\mathbf{s}_{g_0^\dagger}(\mathbf{x}_0) \neq \mathbf{s}_{g_0^\dagger}(\mathbf{x}_1) \ or \ r(\mathbf{x}_0) \neq 0 \quad (6)$$

Namely, the critic learns only when module m_t notices a change of environment or receives actual reward.

3.3 Learning of actors

Each of actors aquires strategy of actions that maximize rewards by corresponding critic in the module. We defined action strategy $\pi(\mathbf{s}, a)$ with which probability of the robot action $a(\mathbf{x})$ on the state $\mathbf{s}_{g^\dagger}(\mathbf{x})$ is calculated as follows.

$$\pi(\mathbf{s}_{g^\dagger}(\mathbf{x}), a(\mathbf{x})) = \Pr\{a(\mathbf{x}) = a \mid \mathbf{s}_{g^\dagger}(\mathbf{x}) = \mathbf{s}\} \quad (7)$$

$$= \frac{\exp(\frac{p(\mathbf{S},a)}{T_a})}{\sum_i \exp(\frac{p(\mathbf{S},i)}{T_a})} \quad (8)$$

Here, $p(\mathbf{s}, a)$ is excellence index of an action a at the state $\mathbf{s}$. Temperature T_a is positive constant that express a ratio of exploration and exploitation.

The actor of selected module learns with estimation error $\epsilon(t)$ by following equation, provided condition (6).

$$p(s, a) \leftarrow p(s, a) + \alpha\epsilon(t) \quad (9)$$

Here, learning rate α is a constant $(0 < \alpha < 1)$.

3.4 Algorithm of module selection

We defined following module selection probability $\Pr(g^\dagger | \mathbf{s}_{\mathbf{G}^*_\mathbf{A}}(\mathbf{x}))$ for position vector $\mathbf{x}$. It maximize total reward of whole model by giving higher selection probability to higher valued module.

$$\Pr(g^\dagger | \mathbf{s}_{\mathbf{G}^*_\mathbf{A}}(\mathbf{x})) = \frac{\exp\left(\frac{\hat{V}_{g^*_{\mathbf{A}_i}}(\mathbf{s}_{g^*_{\mathbf{A}_i}}(\mathbf{x}))}{T_m}\right)}{\sum_j \exp\left(\frac{\hat{V}_{g^*_{\mathbf{A}_j}}(\mathbf{s}_{g^*_{\mathbf{A}_j}}(\mathbf{x}))}{T_m}\right)} \quad (10)$$

Temperature T_m is positive constant that express a ratio of exploration and exploitstion.

4 Experiment on simulation

4.1 Environmental setting and experimental orders

In our simulation, a robot have short and long range sensors which look around its environment in four directions. Learning modules receive the output of sensors. The short range sensors are used to measure distance to nearest wall. Effective range of the short range sensor is from zero to four. If the distance is larger than four, the sensor returns a signal of out of range; five. The long range sensor returns a distance to nearest wall also. Its effective range is from five to nine. If the distance is under five, it returns a signal of too near; four, and if the distance is over nine, it returns a signal of out of range; ten. The robot moves to nearest grid point in eight direction with one action(one step).

In this paper, proposed system is initially evaluated in an environment of Figure 3(a), and after that, the environment is changed to Figure 3(b). We decompose Figure 3(a) environment to situations. The robot get reward r from the environments; plus one for reaching the goal, minus one for colliding a wall and no reward is give otherwise.

4.2 Robustness for environmental change

On the task above, we compared performance of our system with that of conventional system with single module. After learning of these two systems, we

changed the environment by replacing shape of wall at the center of environment for robustness for it.

4.3 Result

For the simulation, parameters C_1, C_2 and C_3 in equation(1) are set to 1.0, 0.5 and 0.9. Maximum number of selected features is set to seven, and the environment is decomposed into 43 situations. In actor, learning rate α and temperature constant T_a are set to 0.2 and 0.1. In critic, learning rate and discount factor γ are set to 0.2 and 0.8.

In the experiment, a robot has learned initial environment for 400 trials, and next 100 trials for changed environment (Figure 2). For the initial environment, success rate of our system rose after 50 trials, and achieved 100% reaching rate by 150 trials. However, conventional system took 100 trials and 250 trial respectively. After the change of environment, success rate of our system kept 100% performance. The reason is that within modules that are necessary to reach goal some of them are not receiveing changed sensory signal and remain unchanged (Figure 3(a)→(b)). On the other hand, the performance of conventional system went down to 0%. This is due to the change of sensory signal at the center of environment, and the learning module could not adapt the change so quickly (Figure 3(c)→(d)).

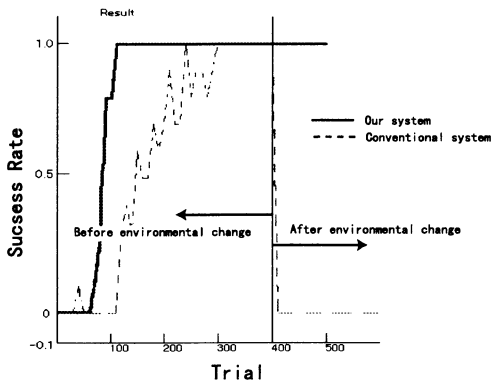


Figure 2: Change of goal reaching performance against environmental change.

5 Conclusion

We proposed a system design concept for learning machine that decomposes environmental situations from sensory data. By computer simulation, we have shown that our system excell learning speed and robustness for environmental change in comparison with conventional system. To examine effectiveness of our concept, we need to apply it to real world signal[5].

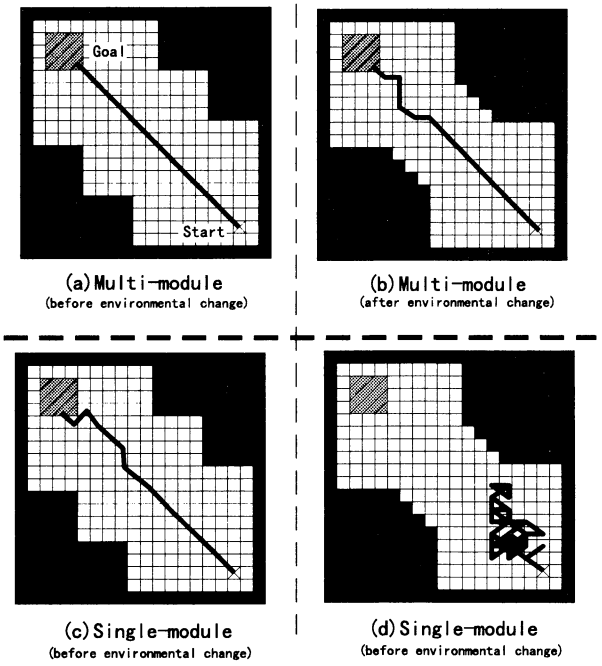


Figure 3: Motion trajectory of robot before and after (401 trial) the environmental change.

References

[1] Breiman, L."Bagging Predictors", Machine Learning, 24, 123-140, 1996

[2] R.Sun, T.Peterson, "Multi-agent reinforcement learning: weighting and partitioning", Neural Networks, 12, 727-753, 1999

[3] Kealbling, L.P.,Littman, M.L.and Moore, A.W, "Reinforcement Learning : A survey", Journal of Artificial Intelligence Research, Vol.4, pp.237-285, 1996

[4] Hiroshi Yamakawa, (1998). " Proposing Matchability Criterion for Situation Decomposition - Extracting situations each of which contains a rule -," Proc. Int. 1998 Conf. on Neural Information Processing (ICONIP'98), vol.3, pp.514-517.

[5] H.Okada, O.Ito, Y.Hagihara, Y.Niki & T.Omori, "Multilevel Environment for Mobile Robotics Capability Experiments(MEMORABLE)", Journal of the Robotics Society of Japan, Vol.17, No.6, pp.896-904, 1999, (in Japanese)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Multi-agent reinforcement learning applied to a chase problem in a continuous world

Hiroki Tamakoshi
Nara Inst. of Sci. and Tech., Japan.
hiroki-t@is.aist-nara.ac.jp
Takayama 8916-5, Ikoma, Nara, Japan

Shin Ishii
Nara Inst. of Sci. and Tech., Japan.
ATR Human Information Processing Research Labs.

Abstract

Reinforcement learning (RL) is one of the methods for solving problems defined upon multi-agent systems. In realistic worlds, the state is continuous, and agents take continuous actions. Since conventional RL schemes are often defined to deal with discrete worlds, there are difficulties such as the representation of an RL utility function.

In this article, we intend to extend an RL algorithm to make it applicable to continuous world problems. This extension is done by the combination of an RL algorithm and a function approximator. We employ Q-learning as the RL algorithm, and a neural network model called the normalized Gaussian network as the function approximator.

The extended RL method is applied to a chase problem in a continuous world. The experimental result shows that our RL scheme proceeds successfully.

1 Introduction

Human beings have long been considering models of intelligent beings. Recently, we have come to recognize the importance of communication among such beings in order to organize a complex society. We call these beings *agents*, and are interested in a model of a society consisting of several or many agents, namely, a multi-agent system.

In the research field of multi-agent systems, there are mainly two approaches to constructing intelligent agents. One is to design the behaviors of the agents by hand, and the other is to let the agents learn appropriate behaviors. In this article, we take the latter approach. Although agents can act and communicate with each other in a given environment, they do not know the meaning of actions or communication prior to learning. They learn the meaning through their experiences in interacting with the environment.

In this study, we consider a multi-agent cooperation problem, in which two agents work together. To let agents learn how to cooperate, we use a *reinforcement learning* (RL) scheme. In RL schemes, an agent learns how good or bad a taken action is through experiences. Since RL schemes were originally developed in single-agent environments, however, the convergence of learning in multi-agent environments has not yet been theoretically guaranteed. On the other hand, a lot of case studies have shown encouraging results for the applicability of RL to multi-agent environments.

In realistic worlds, the state space is continuous, and an agent takes a continuous action in the space. Conventional RL schemes, however, assume that both state and action spaces are discrete, and therefore applications of such RL schemes to continuous world problems involve difficulties. One is the representation of the RL utility function. In order to efficiently represent this function, we employ a *normalized Gaussian network* (NGnet) as a function approximator in this study. The NGnet is a kind of three-layered neural network. This model softly partitions the input space using normalized Gaussian functions, and each local unit approximates the output within its partition. Since the task considered in this study has a spatially neighboring structure, the NGnet is suitable for approximating the utility function.

In order to stabilize learning, we conduct an annealing procedure for the learning coefficient. The merit of the annealing procedure is experimentally confirmed in this study.

2 Reinforcement learning scheme

2.1 General preliminary

Before introducing our reinforcement learning scheme, we need to prepare the notations. Let us first consider a single agent.

s := an environment state that the agent perceives
a := an action that the agent takes
S := the set of s
A := the set of a
$\pi : S \mapsto A$:= a policy function that determines an action for each environment state
r := a reward ($\in \mathbb{R}$)
$R : S \mapsto \mathbb{R}$:= a reward function that determines a reward for each state

It is assumed that the agent is able to perceive the environment s around him. Then, the agent takes an action a which is determined by the policy function π. It is also assumed that a reward r is given to the agent. The reward represents how good or bad the state is; a larger reward implies a better state.

The agent interacts with the environment as follows. (i) The agent takes an action a determined by the policy function π for the current state s, (ii) then the agent interacts with the environment, perceives the next state s', and gets a reward r generated by the reward function R for the state s'. The processes (i) and (ii) are repeated until the agent achieves a goal. The purpose of the agent is to maximize the summation of rewards. The reward function is defined such that the reward summation is large when the agent achieves a goal in an efficient manner.

2.2 Q-learning

Our learning scheme is based on Q-learning[1, 2].

In Q-learning, an action-value function, or a Q-function, $Q(s, a)$, is defined:

$Q(s, a)$:= the estimated summation of rewards if an action a is taken at a state s, and the subsequent actions are determined by the best policy.

If the optimal action-value function $Q^*(s, a)$ is known, the agent can determine the optimal action by $\arg\max_a Q^*(s, a)$ for any state s.

Q-learning is simply stated as follows: each time a state transition occurs from s to s' by taking an action a and a reward is given by $r = R(s')$, then the following renewal of $Q(s, a)$ takes place:

$$Q(s,a) \leftarrow (1-\eta)Q(s,a) + \eta\left\{r + \beta \max_{a'} Q(s',a')\right\}, \quad (1)$$

where η and β are a learning coefficient and a discount factor, respectively.

The update rule (1) influences only a single state-action pair, (s, a). Since the number of possible state-action pairs is infinite in a continuous state-action space, the Q-learning algorithm employing a Q-function defined for a discrete state-action space becomes intractable in computation. In order to overcome this difficulty, a function approximator, for instance, a neural network, is used to represent a Q-function defined for the continuous state-action space. Here, we use the NGnet to approximate the Q-function.

2.3 NGnet and learning algorithm

The NGnet[3] is a network of normalized Gaussian functions. The model softly partitions the input space, and the output is given as a summation of the values given for the partitions. The model is described as follows:

$$\boldsymbol{y} = \sum_{i=1}^{M} \mathcal{N}_i \boldsymbol{w}_i = \sum_{i=1}^{M} \left(\frac{g_i(\boldsymbol{x})}{\sum_{k=1}^{M} g_k(\boldsymbol{x})} \right) \boldsymbol{w}_i \quad (2)$$

$$g_i(\boldsymbol{x}) = \frac{1}{(2\pi)^{\frac{N}{2}} \sigma_i^N} \exp\left(-\frac{|\boldsymbol{x} - \boldsymbol{\mu}_i|^2}{2\sigma_i^2} \right). \quad (3)$$

There are M normalized Gaussian units. $\boldsymbol{x}$ is an N-dimensional input vector, and $\boldsymbol{y}$ is a D-dimensional output vector. g_i is the i-th Gaussian function which has an N-dimensional center vector $\boldsymbol{\mu}_i$ and a variance σ_i^2, and $\boldsymbol{w}_i$ is a D-dimensional vector. The model parameters are $\boldsymbol{\theta} = \{\boldsymbol{\mu}_i, \sigma_i^2, \boldsymbol{w}_i | i = 1, \cdots, M\}$.

We use a *gradient descent* method to let the NGnet learn the relation between the input $\boldsymbol{x}$ and the target output $\boldsymbol{d}$.

$$E \equiv \frac{1}{2}|\boldsymbol{d} - \boldsymbol{y}|^2 = \frac{1}{2}\sum_{j=1}^{D}(d_j - y_j)^2 \quad (4)$$

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \rho \frac{\partial E}{\partial \boldsymbol{\theta}} \quad (5)$$

E is the squared error between the NGnet output $\boldsymbol{y}$ and the target output $\boldsymbol{d}$. ρ is a learning coefficient. Each partial differentiation is calculated as follows:

$$\frac{\partial E}{\partial \boldsymbol{\mu}_i} = \frac{(\boldsymbol{x} - \boldsymbol{\mu}_i)\mathcal{N}_i}{\sigma_i^2} \sum_{j=1}^{D}(y_j - d_j)(w_{ij} - y_j) \quad (6)$$

$$\frac{\partial E}{\partial \alpha_i} = \frac{\mathcal{N}_i}{2}\left(|\boldsymbol{x} - \boldsymbol{\mu}_i|^2 - \frac{N}{\alpha_i} \right) \sum_{j=1}^{D}(d_j - y_j)(w_{ij} - y_j) \quad (7)$$

$$\frac{\partial E}{\partial w_{ij}} = (y_j - d_j)\mathcal{N}_i, \quad (8)$$

where $\alpha_i \equiv \sigma_i^{-2}$.

2.4 RL algorithm with NGnet

The Q-function $Q(s,a)$ defined on the continuous state-action space is approximated by the NGnet. After each transition of the agent, the NGnet is trained using the target output (1) for the input (s,a).

We also engage an annealing procedure for the learning coefficient η of Q-learning. The annealing procedure somehow forms a stochastic approximation. In the experiment, the learning coefficient reduces exponentially: $\eta_t \equiv \gamma\eta_{t-1}$, where t counts the learning episodes, and $0 < \gamma < 1$ is a decay constant.

3 Result

3.1 Task definition

We consider a chase problem in a continuous world, which is a natural extension of the problem defined in a discrete space[4]. The world is a 2D plane and has conjunctive edges.

The system consists of three agents; two chasers, A and B, cooperatively move to attack a prey from both sides. Each agent is assumed to have no volume in the space, i.e., it is just a point. The prey does not learn anything, but has an a priori simple strategy. At each discrete time step, it randomly moves to another position that is limited within a unit circle around the current position, avoiding approaching the chasers. Each chaser acts as follows: (1) it perceives the relative positions of the prey and the other chaser with respect to itself, which is represented as $s \equiv \{s^p, s^c\}$. s^p and s^c denote two-dimensional location vectors for the prey and the other chaser, respectively. (2) it takes an action $a = \arg\max_a Q(s,a)$, where a is a two-dimensional vector and is restricted to be $|a| \leq 1$. Namely, at each time step, each agent is allowed to move within a unit circle around the current position. When the agent takes an action, a small Gaussian noise is added to the radial of the selected action. This noise represents the uncertainty of the actual action. (3) According to the actual action, the state changes. After each agent receives a reward corresponding to the new state, its own Q-function is updated based on Q-learning.

There is a movement order in this system. First the prey moves, then chaser A moves, and finally chaser B moves. These processes are repeated until a goal is achieved.

3.2 Goal state and reward

The goal and the reward function are defined as follows. We consider a unit circle to be a "capture zone" whose center is the current prey position.

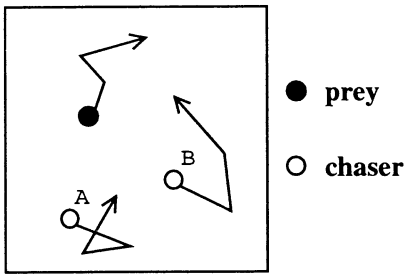


Figure 1: The experiment system

When one or both of the chasers are not in the capture zone, there is no possibility of capturing the prey, and the chasers receive a reward of -0.2. If both of the chasers are in the capture zone, there are two cases, as described below.

Let D_{xy} denote the distance between two agents x and y, where x (or y) is p, A, or B for the prey, chaser A, or chaser B, respectively. If $D_{pA} < D_{AB}$ and the angle between the vector from A to the prey and the vector from A to B is less than $\pi/6$ (case 1), there is a possibility of capturing the prey. This sub-zone in the capture zone is shown as a gray area in figure 2. If this condition is not satisfied (case 2), the reward is -0.1. For case 1, there are two sub-cases. At the possibility $\exp\{-(D_{pA} + D_{pB})\}$, the goal is achieved, and a reward of 1.0 is given to both chasers (case 1-1). If the goal is not achieved, the reward is 0.0 (case 1-2). If case 1-1 occurs, a single episode ends; otherwise it continues.

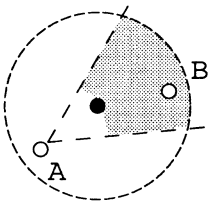


Figure 2: zone with capture possibility

3.3 Solving $\arg\max_a Q(s,a)$

In our learning scheme, it is difficult to obtain $\arg\max_a Q(s,a)$, because there are infinite action candidates for the current state s. Several methods have been proposed[5] to obtain a semi-optimal action from infinite candidates. We use the following method in this study.

For an action candidate $a_0(=(0,0)) \in A$, we prepare a circle in the action space. The center and radius of the circle are a_0 and $I_0(=0.95)$, respectively. The circle is partitioned into mesh grids, and the Q-function is evaluated on the grids. The grid point on

which the Q-function value is the largest is the next candidate a_1. In the next phase, we prepare a smaller circle whose center is a_1 and radius is $I_1 = 0.05I_0$. This process is repeated until the radius I_n becomes small enough. The obtained a_n is the action that the agent takes. Note that this method only obtains a locally optimal action.

3.4 Experimental results

The experiment was conducted under the following conditions.

1. World size: 7×7.
2. Q-learning parameters: $\eta = 0.1$, $\beta = 0.9$, and annealing coefficient: $\gamma = 0.996$.
3. NGnet parameters: $M = 50$, and $\rho = 0.1$

Figure 3 shows the steps to the goal against the number of training episodes. In each episode, the initial states of the agents are randomly prepared in the world. The solid and dashed lines denote the results with and without the annealing process, respectively. Each line denotes the steps averaged over ten learning episodes. We conducted five learning trials by varying the initial learning conditions, and Figure 3 shows their average. This result shows that both agents successfully learned appropriate behaviors.

Figure 4 shows a single learning trial without the annealing. As this figure shows, the learning often becomes unstable without the annealing process. Accordingly, it was found that the annealing procedure has the ability to stabilize the learning.

4 Concluding remarks

Our study is basically an extension of a former study dealing with a discrete state-action space[4].

There have been several studies that extend state or action space to a continuous one[6, 7, 5]. Their methodology for representing the continuous state space, however, is based on a fine discretization of the space, or on an interpolation over the discrete representation of the space. On the other hand, our study intends to deal with the essential continuity using a function approximator.

We introduced a Q-learning scheme that employs an NGnet as a function approximator, which led to a successful result in the experiment. We consider our method to be appropriate for dealing with multi-agent problems that have continuous state and action spaces.

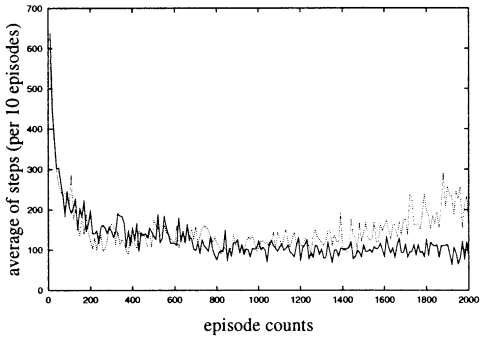


Figure 3: Averaged learning process

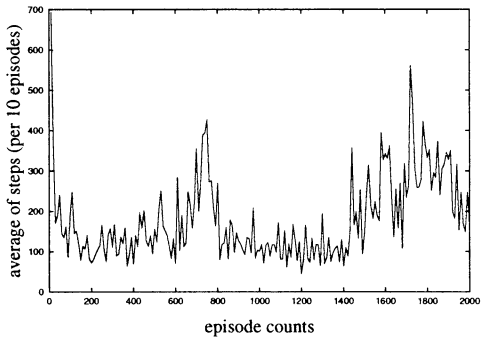


Figure 4: A learning process without annealing

Tan[4] examined the effect of communication between agents, which is an important aspect in multi-agent systems. For human beings, communication is used not only to transfer information, but also to model and predict the behaviors of the partners. We will consider such an aspect in future studies.

References

[1] R. S. Sutton & A. G. Barto. Reinforment Learning: An Introduction. MIT Press, 1998.

[2] C. J. C. H. Watkins and P. Dayan. *Machine Learning Journal*, 8(3/4), 1992.

[3] J. Moody and C. J. Darken. *Neural Computation*, 1(2):281–294, 1989.

[4] M. Tan. In *Proceedings of Tenth International Conference on Machine Learning*, 330–337, 1993.

[5] L. C. Baird and A. H. Klopf. Technical Report WL-TR-93-1147, Wright-Patterson Air Force Base Ohio: Wright Laboratory, 1993.

[6] M. E. Harmon and L. C. Baird. Technical Report WL-TR-1065, Wright-Patterson Air Force Base Ohio: Wright Laboratory, 1996.

[7] R. Munos. In *Proceedings of the 10th European Conference on Machine Learning*, volume 1398 of *LNAI*, 394–405, 1998.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Intelligent Control of Inverted Pendulum Using LQR and Neural Network

Toru Fujinaka and Sigeru Omatu
Dept. of Computer and Systems Sciences
Osaka Prefecture University
Sakai, Osaka, 599-8531 Japan
fuji@cs.osakafu-u.ac.jp

Abstract

An intelligent control which incorporates a combination of linear quadratic optimal regulator and self-tuning neuro-PID control is considered and applied to the control of an inverted pendulum. The former is used when the pendulum is close enough to the equilibrium point, while the latter is used in the other situation. It is shown through simulation studies that such a combination leads to the improvement of the overall performance.

Key words: intelligent control, linear quadratic regulator, neuro-PID control, inverted pendulum.

1 Introduction

Linear quadratic optimal regulator has been widely used as a standard means for controlling linear time-invariant systems. However, it may not be so effective if the controlled plant has nonlinear characteristics or if the system parameters are subject to change. Controllers based on a neural network can cope with the nonlinearities and unknown plant parameters[1]–[3]. However, it may not be easy to guarantee the transient performance the system using the neural network in general.

We consider an intelligent control scheme which uses a combination of the linear quadratic regulator and the neural network controller in order to achieve a better overall control performance. The control of a cart and an inverted pendulum is chosen as a particular example.

The idea is to use the linear quadratic control when the pendulum is sufficiently close to its inverted position, i.e. the equilibrium point. The neural network based controller is used in the other situation, especially in the initial operation of swinging up the pendulum, where the pendulum is not quite close to the equilibrium point. Among the various type of controllers using a neural network, we adopt the self-tuning neuro-PID controller[4] which has been known to be capable of controlling the inverted pendulum in a wide range of angular position.

When the pendulum reaches a certain range of angular position, using the neuro-PID control, the control is switched to linear quadratic regulator. Then the linear quadratic regulator brings the pendulum to the upper equilibrium point with a better transient response than those achieved by the neuro-PID controller. If the pendulum position deviates beyond the region covered by the linear quadratic regulator for some reason, e.g. by some disturbance, the control is switched back to the neuro-PID.

2 Self-Tuning Neuro-PID

In real control processes, the PID controller has been used as a major control method. The main reason is that its structure is simple and PID controllers are robust with noise and parameter variations. Usually, the operators select suitable PID gains from time to time based on their experience and knowledge about the plant dynamics. However, it is desirable to find a suitable set of PID gains automatically. In this paper we use a neural network for tuning the PID gains and apply it to the control of an inverted pendulum.

Among many neural network learning methods, we adopt the error back-propagation(BP) method[1] which is widely used for the training of various kinds of neuro-controllers. The basic control scheme is depicted in Fig. 1, where feedforward controller (FFC) and feedback controller (FBC) are adjusted.

The neuro-controller is used in FFC or FBC instead of a conventional controller. Among the typical neuro-controllers of series type, parallel type, and self-tuning

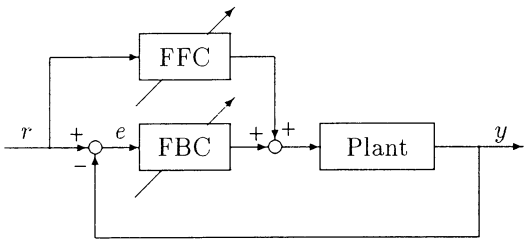


Fig. 1 General control scheme

type, we consider the self-tuning type in which the FFC is neglected and the PID structure is adopted for FBC.

The control structure of the self-tuning neuro-PID controller is shown in Fig. 2, where the outputs of the neural network are proportional (P), integral (I), and derivative (D) gains, and the inputs are selected in a suitable way according to the specific problem.

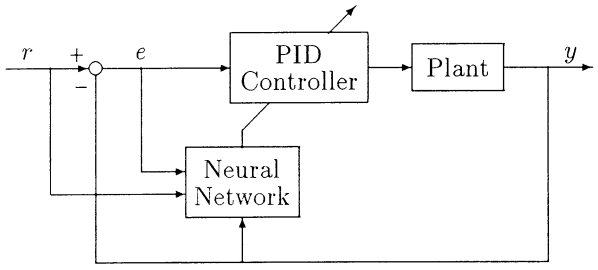


Fig. 2 Self-tuning neuro-PID control scheme

In order to derive the self-tuning neuro-PID control algorithm, we will define mathematical notation of neuron models. Total input to neuron j is denoted by net_j and is given by

$$net_j = \sum_{i=1}^{n} w_{ji} O_i + \theta_j$$

where θ_j denotes a threshold. Then the output O_j of neuron i becomes

$$O_j = f(net_j), \quad f(x) = \frac{1}{1+e^{-x}}$$

By connecting many neurons in a layered array, we can derive the BP algorithm based on the gradient method which is described as follows:

Step 1 Find the generalized error δ_k in the output layer

$$\delta_k = (t_k - O_k) f(net_k) = (t_k - O_k) O_k (1 - O_k)$$

and the generalized error in the hidden layer

$$\delta_j = f(net_j) \sum_{k=1}^{K} \delta_k w_{kj} = O_j (1 - O_j) \sum_{k=1}^{K} \delta_k w_{kj}$$

where t_k is the target pattern, O_k is the actual output of neural networks, and w_{ji} denotes the connection weight from neuron i to neuron j.

Step 2 Change the connection weights by Δw_{ji} in the following way: in the output layer

$$\Delta w_{kj}(n+1) = \eta \delta_k O_j + \alpha \Delta w_{kj}(n) + \beta \Delta w_{kj}(n-1)$$

and in the hidden layer

$$\Delta w_{ji}(n+1) = \eta \delta_k O_j + \alpha \Delta w_{ji}(n) + \beta \Delta w_{ji}(n-1)$$

where a modified BP method has been used to avoid the local minimum problem.

Step 3 If the error becomes smaller than a predetermined value, then stop. Otherwise, replace n by $n+1$ and go to Step 1.

Using the above BP algorithm, we derive the self-tuning neuro-PID control algorithm. First, note that the PID controller in discrete-time is described as

$$\begin{aligned} u(n) = u(n-1) & \\ & + K_P(e(n) - e(n-1)) + K_I e(n) \\ & + K_D(e(n) - 2e(n-1) + e(n-2)) \end{aligned} \tag{1}$$

where K_P, K_I, and K_D are proportional, integral, and derivative gains, respectively, $u(n)$ denotes the plant input at nT, and $e(n) := r(n) - y(n)$ is the error between the desired value $r(n)$ and the actual output $y(n)$ at time nT. Here, T denotes the sampling interval.

In order to adjust K_P, K_I, and K_D adaptively, we use a three layered neural network based on the BP method. Each layer consists of N_1, N_2, and N_3 neurons, where N_1 and N_2 can be selected by trial and error according to complexity of the plant, and $N_3 = 3$, corresponding to the number of PID gains. The BP method requires a cost function E which should be minimized. Here, the following E is adopted such that the PID controller can decrease the squared error

$$E := \frac{1}{2} e(t+1)^2$$

at time $(n+1)T$.

The connection weights w_{kj} at the output layer are updated by the equation

$$\Delta w_{kj}(n+1) = \eta\delta_k O_j + \alpha\Delta w_{kj}(n) + \beta\Delta w_{kj}(n-1)$$

where

$$\delta_k = -\frac{\partial E}{\partial net_k}.$$

Using the chain rule for derivative, we get

$$\delta_k = e(n+1)\frac{\partial y(n+1)}{\partial u(n)}\frac{\partial u(n)}{\partial O_k}\frac{\partial O_k}{\partial net_k}$$

Furthermore, we have from (1)

$$\frac{\partial u(n)}{\partial O_k} = \begin{cases} e(n) - e(n-1), & (k=1) \\ e(n), & (k=2) \\ e(n) - 2e(n-1) + e(n-2), & (k=3) \end{cases}$$

where $O_1 = K_P$, $O_2 = K_I$, and $O_3 = K_D$.

If the plant is unknown, two approximation algorithms to calculate the system Jacobian $\partial y(n+1)/\partial u(n)$ have been proposed.

The first one is to use the following relation:

$$\frac{\partial y(n+1)}{\partial u(n)} = \mathrm{sgn}\left(\frac{\partial y(n+1)}{\partial u(n)}\right)\left|\frac{\partial y(n+1)}{\partial u(n)}\right|$$

Thus, if we only know the sign of the Jacobian, we can adjust the learning rate η such that $|\partial y(n+1)/\partial u(n)|$ is included into η.

The second way is to build an emulator which simulates the input/output map of the plant. Regarding the emulator as the true plant and using the BP method, we can adjust the connection weights by combining the neural network to tune the PID gains with a neural network to simulate the plant. The overall system for the second approach is constructed by adding the emulator in parallel to the plant. Using this control scheme, we have obtained fine tuning of the PID gains in many cases.

3 Linear Quadratic Regulator

The linear quadratic regulator has been used as a standard control method, provided the controlled plant can be regarded as a linear time-invariant system. Suppose the plant is given by the following discrete-time state equation

$$x(k+1) = Ax(k) + Bu(k)$$
$$y(k) = Cx(k)$$

and the performance index to be minimized is given by

$$J := \sum_{k=0}^{\infty}\left(x^T(k)Qx(k) + u^T(k)Ru(t)\right)$$

where Q and R are positive-definite weighting matrices. Then the optimal input u is given by the state feedback law of the form

$$u(k) = Fx(k), \quad F := (R + B^T PB)^{-1}B^T PA$$

where P in the optimal feedback gain F is the positive definite solution of the discrete Riccati equation

$$P = A^T PA - A^T PB(R + B^T PB)^{-1}B^T PA + Q.$$

The optimal control is known to be effective, but the plant parameters comprising the coefficient matrices A and B are assumed to be known, and the weighting matrices Q, R, the design parameters, are to be carefully chosen by the designer. Since a fairly good model of the inverted pendulum is obtained and it can be linearized at the equilibrium point, we can apply the LQ regulator to the plant described by the above difference equation.

However, it should be noted that the LQ regulator may not be effective if the pendulum is not close enough to the equilibrium point. Thus, we consider the combination of neural-network based controller and the LQ regulator for the control of inverted pendulum.

4 Control of inverted pendulum

We consider the application of the single-input two-output self-tuning PID controller to stabilization of the inverted pendulum. The control system of the inverted pendulum is described in Fig. 3. Our aim is to move the cart of the pendulum to the center of the bar ($x = 0$) and let the pendulum stand at the upright position ($\theta = 0$).

To construct the PID controller, the PID gains are adjusted by the neural network as shown in Fig. 4, where the minimum cost function for tuning is the squared error between the actual output and the desired values. In this case the desired values are $x = 0$ and $\theta = 0$.

As we need the system Jacobian of the inverted pendulum, we use the value derived by using the emulator mentioned above. The input layer of the emulator consists of six units and the output layer

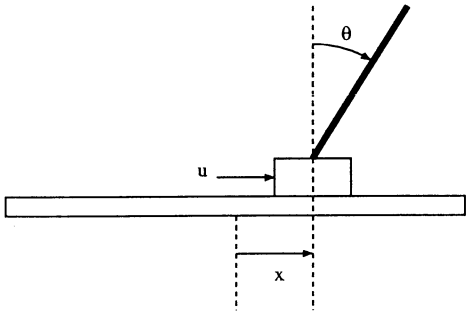


Fig. 3 Inverted pendulum control system

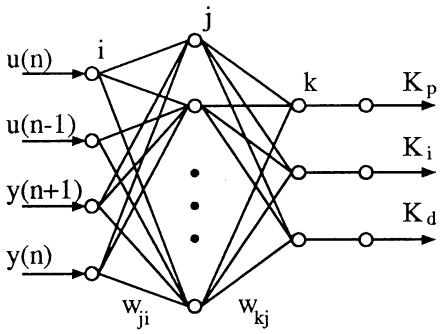


Fig. 4 Neural network for gain tuning

has two units. We obtain the system Jacobian by the following equations:

$$\frac{\partial \hat{x}(n+1)}{\partial u(n)} = \sum_{j=1}^{3} w'_{kj} O_j (1 - O_j) w'_{j1}$$

$$\frac{\partial \hat{\theta}(n+1)}{\partial u(n)} = \sum_{j=4}^{6} w'_{kj} O_j (1 - O_j) w'_{j1}$$

Here, the suffix 1 of w_{j1} denotes the neuron unit corresponding to $u(n)$ at the input layer, and $j = 1, 2, 3$ denote the three neurons in the hidden layer which are connected to $\hat{x}(n+1)$. Similarly, $j = 4, 5, 6$ denote the remaining neurons in the hidden layer which are connected to $\hat{\theta}(n+1)$.

Using the above relations, we have obtained the control results of the inverted pendulum. The control results are shown in Fig. 5, where the position x and angle θ quickly approach and stay close to the equilibrium point.

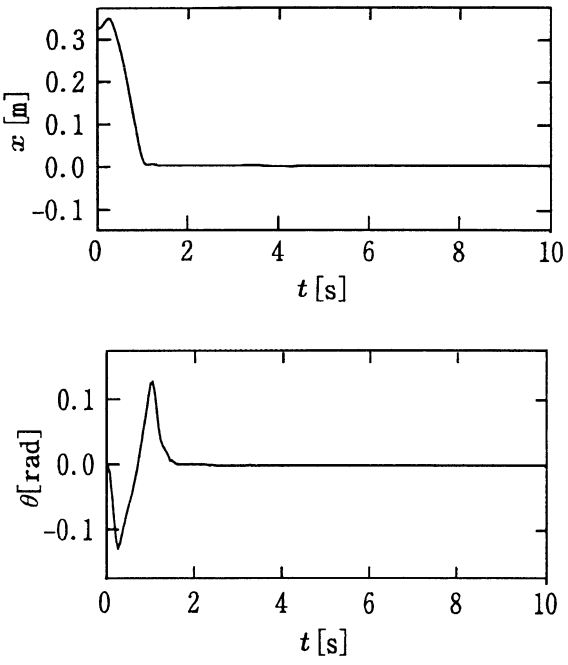


Fig. 5 Control results of inverted pendulum

5 Summary

Intelligent control using linear quadratic regulator and neuro-PID controller is applied to an inverted pendulum. The effectiveness of the proposed method was shown through computer simulation and experiment. Further improvement of the performance may be expected by investigating a better switching scheme between the linear quadratic regulator and the neural network based controller.

References

[1] W. T. Miller III, R. S. Sutton, and P. J. Werbos, *Neural Networks for Control*, MIT Press, 1990.

[2] D. A. White and D. A. Sofge (Eds.), *Handbook of Intelligent Control, Neural, Fuzzy, and Adaptive Approaches*, Van Nostrand Reinhold, 1992.

[3] S. Omatu, M. Khalid, and R. Yusof, *Neuro-Control and Their Applications*, Springer-Verlag, 1995.

[4] T. Aoyama and S. Omatu, "Design of a Self-Tuning PID Control System by Neural Networks", *Trans. on IEE of Japan*, Vol. 116-C, No. 11, pp. 1197–1201, 1996 (in Japanese).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Dynamic Identification of an Industrial Seven-Link Manipulator by Evolving Runge–Kutta-Gill RBF Networks

Thrishanta Nanayakkara*, Keigo Watanabe**, Kazuo Kiguchi** and Kiyotaka Izumi***
*Graduate School of Science and Engineering,
Faculty of Engineering Systems and Technology,
**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
†E-mail: watanabe@me.saga-u.ac.jp

Abstract

This paper proposes a method for identification of dynamics of a multi-link robot manipulator using Runge-Kutta-Gill neural networks (RKGNNs). Shape adaptive radial basis function (RBF) NNs have been employed with an evolutionary algorithm to optimize the shape parameters and the weights of the RKGNN. Unlike in conventional methods, the proposed method can even be used without input torque information because a torque network is part of the functional network. The method has been employed for dynamic identification of a seven-link industrial manipulator called PA-10, manufactured by the Mitsubishi Heavy Industries Ltd.

Keywords:Runge-Kutta-Gill neural networks, Radial basis functions, Multi-link robot manipulators, Evolutionary optimization.

1 Introduction

Stability and accuracy of model based controllers such as computed torque algorithms demands for effective identification of dynamics of manipulators [1]. In most of the industrial robot manipulators, the parameters of the dynamics are vaguely known or incompletely known to design such control algorithms. Identification of inertia parameters using the least squares algorithm has been investigated in [2]. These methods have the problems of finding a globally optimum solution and the difficulty of identifying a complex system with high degrees-of-freedom due to the backpropagation algorithm to find weights of NNs. Moreover these methods use input torque information for the calculations that can be expensive on one hand and on the other hand, needs special hardware changes.

Therefore the motivation of this research has been to develop a method that is capable of identifying the dynamics of a complex multi-link robot manipulator with basic reference inputs of joint angle and its velocity, and their output data.

The proposed method uses Runge–Kutta–Gill neural networks (RKGNNs), which is an extension version of Runge–Kutta neural networks (RKNNs), for the identification of the dynamics, because RKNNs are capable of identifying the changing rates of the states accurately due to the state space interpolation between one sampling interval [3]. Furthermore, this method only needs one state information to predict the next state and the trained NN of the function is independent of the sampling time interval unlike in direct mapping neural networks (DMNN). The function NN has sub networks to represent the components of the dynamics of the manipulator [1]. Radial basis function (RBF) NNs have been used for each sub-network. Due to the complex structure of the dynamics of a manipulator, the number of weights of the total NN system demands for efficient optimization techniques to train the networks. Therefore optimization of weights using a new evolutionary optimization algorithm [4] has been proposed in this paper. Due to the property of monotonically reducing standard deviation of mutation, the convergence rate is fast in the evolutionary algorithm. In the proposed algorithm, sustaining adequate standard deviation of mutation throughout the optimization process guarantees the convergence to a global optimality.

PA-10 manipulator with seven-links has been subjected to the dynamics identification study. Experimental data of basic reference input position, its velocity and their output data have been used to train the RKGNNs.

Rest of this paper is organized as follows: In section 2, a brief introduction to the basic components of the dynamics of a multi-link robot manipulator and the problems in a practical dynamics identification s-

tudy is explained. In section 3, the RKGNNs and the structure of the function NNs are explained. In section 4, the procedure of training the NNs for PA-10 manipulator is explained. In section 5, the results are discussed.

2 Dynamics of a manipulator

Dynamics of a manipulator is given by

$$\boldsymbol{\tau} = M(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + V(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + G(\boldsymbol{\theta}) + F(\boldsymbol{\theta}) \qquad (1)$$

where $\boldsymbol{\tau}$ is the joint torque vector, $\boldsymbol{\theta}$ is the joint angle vector, $\dot{\boldsymbol{\theta}}$, $\ddot{\boldsymbol{\theta}}$ are joint angular velocity and angular acceleration vectors respectively, $M(\boldsymbol{\theta})$ is the inertia matrix, $V(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ is the Coriolis and centrifugal force vector, $G(\boldsymbol{\theta})$ is the gravity force vector, and $F(\boldsymbol{\theta})$ is the friction vector [1].

It can be seen in [1] that the derivation of this dynamic equation for a manipulator with higher degrees of freedom such as PA-10 manipulator, is very complex and time consuming. Even if the dynamic equation is known for such a manipulator, many crucial parameters such as link inertia are unknown or partially known. The identification of these unknown parameters involves data collection including the input torque sensor information if conventional least-squares method [2] or DMNN approach is employed. Yet in most practical situations, the torque sensors are not built-in with the manipulator. Therefore if torque sensors are to be used, some hardware changes are required that makes additional trouble and expense.

The proposed method attempts to identify the dynamics involving only the reference input of joint angle and its velocity, and their output information, using the RKGNNs trained by an evolutionary algorithm. This needs rearranging of the dynamic equation in (1) to form an ordinary differential equation (ODE) of the form

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}). \qquad (2)$$

Given the dynamics of the manipulator as in equation (1), the state vector $\boldsymbol{x}$ can be defined by

$$\boldsymbol{x} = [\boldsymbol{\theta} \;\; \dot{\boldsymbol{\theta}}]^T. \qquad (3)$$

Then the ODE given by (2) can be formed using $\boldsymbol{x}$ and the dynamics defined by equation (1) as

$$\begin{bmatrix} \dot{\boldsymbol{\theta}} \\ \ddot{\boldsymbol{\theta}} \end{bmatrix} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta} \\ \dot{\boldsymbol{\theta}} \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1}(\boldsymbol{\theta})\{\boldsymbol{\tau} - V(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})...\} \end{bmatrix}$$

The right-hand side of the above ODE gives the rate of change of the states given by the left-hand side. Therefore for this family of ODEs, RKGNNs may very accurately identify the right hand side.

3 RKGNNs and the structure of the function networks.

Figure 1 shows the basic RKGNN algorithm. From Fig. 1, it can be seen that the prediction of the next state involves one previous state information and the sampling interval is external to the function NN model. One other feature is that all four NN models given by N_f in the RKGNN are the same. Therefore finding the optimum weights for one N_f is all that is required to identify the system. The RKGNN algorithm contains the following steps.

Step 1:

$$\boldsymbol{k}_0 = hN_f(\boldsymbol{x}(i))$$

$$\boldsymbol{x}^{(1)}(i+1) = \boldsymbol{x}(i) + \frac{1}{2}\boldsymbol{k}_0$$

$$\boldsymbol{q}_1 = \boldsymbol{k}_0$$

Step 2:

$$\boldsymbol{k}_1 = hN_f(\boldsymbol{x}^{(1)}(i+1))$$

$$\boldsymbol{x}^{(2)}(i+1) = \boldsymbol{x}^{(1)}(i+1) + \alpha(\boldsymbol{k}_1 - \boldsymbol{q}_1)$$

$$\boldsymbol{q}_2 = \beta\boldsymbol{q}_1 + e\boldsymbol{k}_1$$

Step 3:

$$\boldsymbol{k}_2 = hN_f(\boldsymbol{x}^{(2)}(i+1))$$

$$\boldsymbol{x}^{(3)}(i+1) = \boldsymbol{x}^{(2)}(i+1) + d(\boldsymbol{k}_2 - \boldsymbol{q}_2)$$

$$\boldsymbol{q}_3 = \gamma\boldsymbol{q}_2 + g\boldsymbol{k}_2$$

Step 4:

$$\boldsymbol{k}_3 = hN_f(\boldsymbol{x}^{(3)}(i+1))$$

$$\boldsymbol{x}^{(4)}(i+1) = \boldsymbol{x}^{(3)}(i+1) + \frac{1}{6}\boldsymbol{k}_3 - \frac{1}{3}\boldsymbol{q}_3$$

$$\boldsymbol{x}(i+1) \equiv \boldsymbol{x}^{(4)}(i+1)$$

In this case, h is the time step width, $d = (\sqrt{2}+1)/\sqrt{2}$, $e = (2-\sqrt{2})$, $g = (2+\sqrt{2})$, $\alpha = (\sqrt{2}-1)/\sqrt{2}$, $\beta = -2+3/\sqrt{2}$, and $\gamma = -(2+3/\sqrt{2})$.

The function NN $N_f(\cdot)$ as shown in **Fig. 2** contains components for the identification of each component of the dynamic equation as shown in equation (1). Since information about the controller is unknown, a torque network is constructed that will grasp the dynamics of the controller.

The details of the component networks of inertia, velocity including the Coriolis and centrifugal terms, and gravity are presented in [5]. The construction of

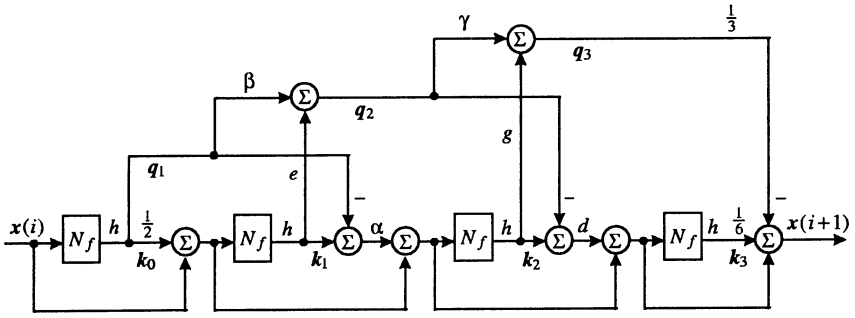


Figure 1: Structure of the fourth-order RKGNN by the function neural networks N_f.

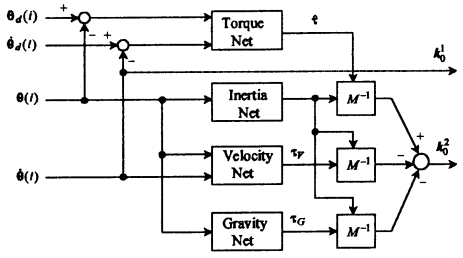


Figure 2: Basic concept of the function NN to identify each component of the dynamic equation.

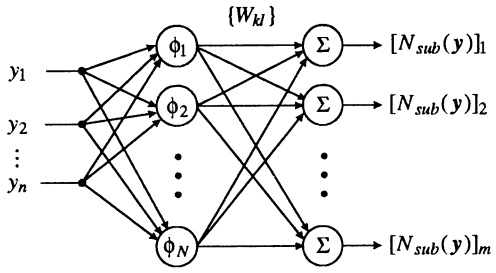


Figure 3: Basic structure of the RBF NN.

sub-networks of each of these component networks involve construction of shape adaptive RBF NNs as can be seen in **Fig. 3**.

The input-output mapping relationship of the RBF in this application is given by

$$\phi_l(\boldsymbol{y}; \bar{\boldsymbol{y}}_l, \sigma_l) = \exp\{-[(\boldsymbol{y} - \bar{\boldsymbol{y}}_l)^T(\boldsymbol{y} - \bar{\boldsymbol{y}}_l)/\sigma_l^2]\} \quad (4)$$

where $\boldsymbol{y} \in \Re^n$ and $\bar{\boldsymbol{y}}_l \in \Re^n$ are the input vector and the vector of centers of the radial basis function, σ_l^2 is the varaince of the lth RBF. Therefore the output at the kth output node of the sub-network $N_{sub}(\cdot)$ is given by

$$[N_{sub}(\boldsymbol{y})]_k = \sum_{l=1}^{N} W_{kl}\phi_l(\boldsymbol{y}; \bar{\boldsymbol{y}}_l, \sigma_l), k = 1, ..., m \quad (5)$$

where N is the number of RBFs that construct the sub-network and W_{kl} is the weight from lth RBF to kth output node.

4 Identification of dynamics of PA-10 manipulator

The PA-10 manipulator consists of seven links with seven degrees of freedom (three rotation axes and four pivot axes) as illustrated in **Fig. 4**. The main specification of the PA-10 manipulator are as follows: the length of the manipulator is 1345 mm, the weight of the manipulator is 35 kgf, the maximum combined speed with all axes is 1.55 m/s, the payload weight is 10 kgf, and output torque is 9.8 Nm. The input reference angle, its velocity and their output data of all seven joints were experimentally obtained. Then these experimental data were used to train the RKGNNs to identify the dynamics of the manipulator. The weights of the RKGNN and the RBF parameters, $\bar{\boldsymbol{y}}_l$ and σ_l, were found using an evolutionary algorithm so as to minimize the cost function given by

$$J = \frac{1}{t_N}\sum_{i=1}^{t_N} ||\boldsymbol{\theta}_d(i) - \boldsymbol{\theta}(i)|| + ||\dot{\boldsymbol{\theta}}_d(i) - \dot{\boldsymbol{\theta}}(i)|| + P \quad (6)$$

where t_N is the total time span of the trajectory, P is a penalty. If the minimum of the principle minors of the inertia matrix $M(\boldsymbol{\theta})$ estimated by the network is negative, P is given the negative value of the minimum of the principle minors of the inertia matrix $M(\boldsymbol{\theta})$ estimated by the network. Otherwise P is set to zero.

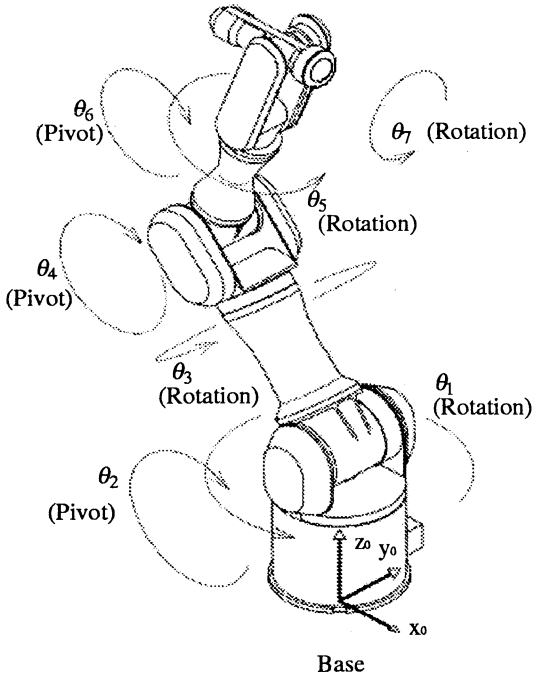


Figure 4: Mechanical structure of the PA-10 manipulator.

The reason to add this penalty is to make sure that the estimated inertia matrix is symmetric and positive definite. The conditions of the usage of evolutionary algorithm were as follows: 100% mutation and 20% crossover was employed. The tournament size was 15 and the number of individuals were 100. One individual consists of 3080 elements. The algorithm was run for 200 generations.

5 Results and discussion

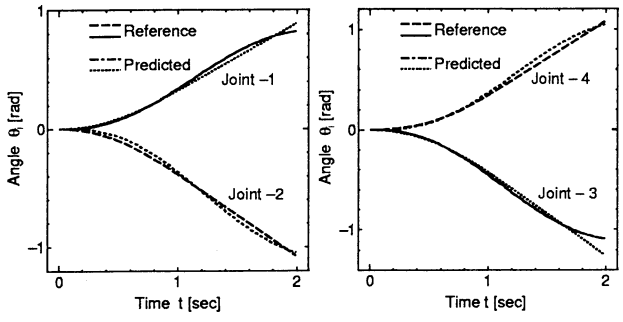


Figure 5: The reference and predicted joint angular trajectories. On the left is that for joint 1 and 2, and on the right is that for joint 3 and 4.

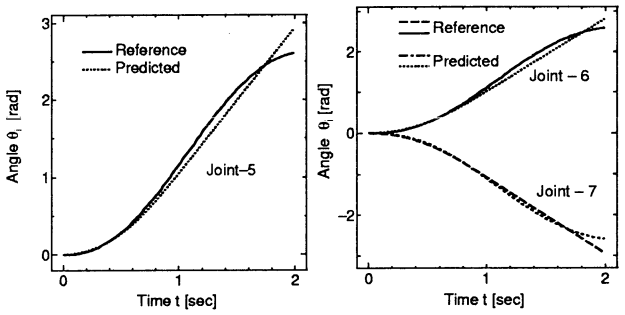


Figure 6: The reference and predicted angular trajectories. On the left is that for joint 5, and on the right is that for joint 6 and 7.

The RKGNN has been developed and it was applied for identifying the complex dynamics of a seven-link industrial manipulator called PA-10, only using the input angle, its velocity and their output data. It is very useful, because it reduces the burden of deriving the complex dynamic equations for such manipulators. The adoption of RKGNNs makes the long term prediction of the states very accurate due to its property of interpolation of states within one sampling interval and estimating the rate of change of states accurately. This method is advantageous in the practical situation where input torque information is unknown.

References

[1] J. J. Craig, *Introduction to Robotics: Mechanics and control,* 2nd edn, Reading, MA: Addison-Wesley, 1989.

[2] C. H. An, C. G. Atkeson, J. M. Hollerbach, "Estimation of Internal Parameters of Rigid Body Links of Manipulators," *Artificial Intelligence Memo 887, MIT Artificial Intelligence Laboratory,* 1986.

[3] Y.-J. Wang and C.-T. Lin, "Runge-Kutta Neural Network for Identification of Dynamical Systems in High Accuracy," *IEEE Trans. on Neural Networks,* vol. 9, no. 2, pp. 294–307, 1998.

[4] T. Nanayakkara, K. Watanabe and K. Izumi, "Evolving in Dynamic Environments Through Adaptive Chaotic Mutation," in *Proc. of Fourth International Symposium on Artificial Life and Robotics,* vol. 2, 1999, pp. 520–523.

[5] T. Nanayakkara, K. Watanabe, and K. Izumi, "Evolving Runge–Kutta–Gill RBF Networks to Estimate the Dynamics of a Multi-Link Manipulator," in *Proc. of IEEE International Conference on Systems, Man, and Cybernetics,* vol. 2, 1999, pp. 770–775.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Control of Robot Manipulators via Chaotic Attractors and Fuzzy Model-Based Regulators

Lanka Udawatta*, Keigo Watanabe**, Kazuo Kiguchi** and Kiyotaka Izumi*
*Department of Mechanical Engineering,
Faculty of Science and Engineering,
**Department of Advanced Systems Control Engineering,
Graduate School of Science and Engineering,
Saga University, 1-Honjomachi, Saga 840-8502, Japan
†E-mail : watanabe@me.saga-u.ac.jp

Abstract

A Chaotic attractor can be formed in order to employ an open-loop control scheme to a dynamical system until the system states converge within a specified area of the attractive domain. In this approach, first phase uses open-loop control forming a chaotic attractor. Fuzzy model-based controller is employed under state feedback control (SFBC) in the second phase of control. The relaxed stability conditions and LMI (Linear Matrix Inequalities) based designs for a fuzzy regulator are used to construct the fuzzy attractive domain, in which a global solution is obtained so as to achieve the desired stability condition of the closed-loop system. A two-link robot manipulator is employed to illustrate the above design procedure of the proposed controller.

Keywords: Chaotic systems, Nonholonomic systems, Free joints, Nonlinear dynamics, Fuzzy model-based control.

1 Introduction

During the last two decades, development of chaos theory, which brings up researchers into a new era in studying nonlinear systems which, exhibits a deterministic random behavior. Employing intended deterministic chaos to control nonlinear dynamical systems is an interesting in developing control algorithms [1], [6]. In this method, it is proposed that in a nonlinear dynamical system, a chaotic attractor can be formed by an appropriate usage of open-loop control (OLC) until the system states converge within a specified area of attractive domain and then SFBC can be applied to drive the system states to desired values. The drawback of this approach is eliminated by Watanabe *et al.* [9] as proposed in the control of chaotic systems using fuzzy model-based regulators. Specially, controlling of robot manipulators with the aid of chaos phenomena has been presented in [8], [10]. In [8], nonlinear behavior of manipulators with free-joints and control techniques via averaging method are explained.

Recent advances in LMI theory allowed handling nonlinear control system problems via semi-definite programming [2],[3],[4]. In particular, Takagi-Sugeno (T-S) fuzzy model plays an important role in designing such fuzzy regulators [3],[4],[5],[7]. In fact, the fuzzy model-based control (FMC) can be applied very well to nonlinear dynamical systems, this attempt also implies the ability of controlling chaotic systems via FMC. In addition to that, the proposed method has the advantage of solving LMIs either using evolutionary computation as proposed in [9] or recently developed powerful semi-definite programming tools based on convex optimization algorithms available to date in mathematical literature.

In the control of chaotic systems which have inherent chaotic attractive features or created chaos, there are two phases of control [9]. First phase uses an open-loop controller as studied in [1],[9]. Once the system has entered a specified area, OLC is cut off and the proposed second phase of control scheme is adopted. In the proposed second phase, an FMC is employed under SFBC. For this purpose, the state feedback gain scheduling of the control system in the second phase is achieved by solving a set of LMIs via an optimization technique based on evolutionary computation. This method leads to eliminate the shortcomings of the approach [1] while employing useful chaotic features. The rest of the paper is organized as follows: In Section 2, concept of the controller and FMC are reviewed. A two-link robot manipulator is taken into consideration so as to illustrate the design procedure and simulation results are presented in Section 3.

2 Implementation of the Controller

2.1 Concept of Proposed Controller

As proposed in [9], chaotic features can be used to drive the system states to a pre-defined domain C. An appropriate OLC input can be employed to create chaos in a nonlinear dynamical system. Once it reaches to the pre-defined fuzzy attractive domain, a FMC is employed under SFBC to achieve desired target. This design concept is schematically given in **Fig.**

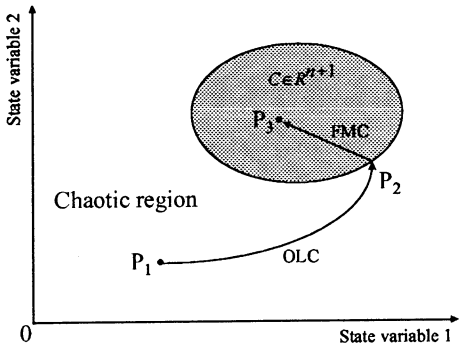


Figure 1: Concept of fuzzy-chaos hybrid control scheme

1 for a two dimensional case. Here it is intended to drive the system state P_1 to P_3.

2.2 Fuzzy Models as Applied to SFBC

Fuzzy models due to TS consist of a set of IF-THEN rules [5] for the above approximate systems. The ith plant rule of each subsystems for a continuous-time systems is given by

$$\text{IF} \quad z_1(t) \text{ is } M_{i1} \text{ and...and } z_p(t) \text{ is } M_{ip}$$
$$\text{THEN} \quad \begin{cases} \dot{\boldsymbol{x}}(t) &= A_i \boldsymbol{x}(t) + B_i \boldsymbol{u}(t) + \boldsymbol{d}_i \\ \boldsymbol{y}(t) &= C_i \boldsymbol{x}(t) \quad i = 1, ..., r \end{cases}$$

where r is the number of fuzzy rules and M_{ij} ($i = 1, ..., r$ and $j = 1, ..., p$) are the fuzzy sets. The state vector is $\boldsymbol{x}(t) \in \Re^n$, input vector is $\boldsymbol{u}(t) \in \Re^m$ and the output vector is given by $\boldsymbol{y}(t) \in \Re^q$. A_i, B_i and C_i are the system parameter matrices and $\boldsymbol{d}_i$ is the offset term of the ith fuzzy model. For a given state, $z_1(t), ..., z_p(t)$ are the premise variables (or antecedent inputs).

Subjecting to the parallel distributed compensation, we can design the following fuzzy regulators:

Regulator Rule i :

IF $z_1(t)$ is M_{i1} and...and $z_p(t)$ is M_{ip} THEN

$$\boldsymbol{u}(t) = -K_i[\boldsymbol{x}(t) - \boldsymbol{x}_r] + \boldsymbol{u}_r$$

where $\boldsymbol{x}_r$ is a state reference trajectory, $\boldsymbol{u}_r$ is the corresponding input trajectory and K_i is the local feedback gain matrix. Thus, the fuzzy regulator rules have linear state-feedback laws in the consequent parts and the overall fuzzy regulator can be reduced to

$$\boldsymbol{u}(t) = -\sum_{i=1}^{r} h_i(\boldsymbol{z}(t)) K_i[\boldsymbol{x}(t) - \boldsymbol{x}_r] + \boldsymbol{u}_r$$

where

$$\boldsymbol{z}(t) = [z_1(t), ..., z_p(t)]$$
$$w_i(\boldsymbol{z}(t)) = \prod_{j=1}^{p} M_{ij}(z_j(t))$$

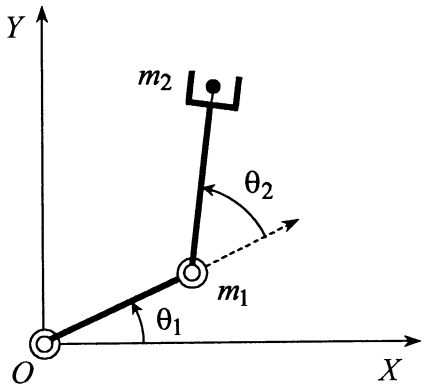


Figure 2: Two-link manipulator

$$h_i(\boldsymbol{z}(t)) = \frac{w_i(\boldsymbol{z}(t))}{\sum_{l=1}^{r} w_l(\boldsymbol{z}(t))}$$

for all t, in which $M_{ij}(z_j(t))$ denotes the confidence (or grade) of membership of $z_j(t)$ in M_{ij}.

2.3 Total Controller

In order to control the original nonlinear system with a chaotic input in the open-loop system or a fuzzy controller in the closed-loop system, the fuzzy-chaos hybrid control scheme can be logically implemented as follows:

$$\begin{aligned} &\text{IF} \quad \sum_{i=1}^{r} w_i(\boldsymbol{z}(t)) \equiv 0 \quad \text{THEN} \\ &\quad \boldsymbol{u}(t) = \hat{\boldsymbol{u}}(t) \\ &\text{ELSE} \\ &\quad \boldsymbol{u}(t) = -\sum_{i=1}^{r} h_i(\boldsymbol{z}(t)) K_i[\boldsymbol{x}(t) - \bar{\boldsymbol{x}}] + \bar{\boldsymbol{u}} \end{aligned} \tag{1}$$

where $\hat{\boldsymbol{u}}(t)$ is an open-loop input to make the original nonlinear system chaotic and $(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}})$ is the prespecified equilibrium point which would be stabilized. Gain scheduling of the feedback controller is determined by solving a set of LMIs [2],[9].

3 Design Example

3.1 Manipulator Modelling

Two-link manipulator in **Fig. 2** is taken into consideration. Joint torque vector $[\tau_1 \ \tau_2]^T$ of the manipulator is given in the dynamic equation (2) as below:

$$\begin{aligned} \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \boldsymbol{M} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} &+ (\dot{\theta}_1 \dot{\theta}_2) \boldsymbol{B} \\ + \boldsymbol{C} \begin{bmatrix} \dot{\theta}_2^2 \\ \dot{\theta}_1^2 \end{bmatrix} &+ \boldsymbol{F} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \boldsymbol{G} \end{aligned} \tag{2}$$

where inertia matrix is given by

$$\boldsymbol{M} = \begin{bmatrix} m_2 l_2^2 + 2m_2 l_1 l_2 c_2 + (m_1 + m_2) l_1^2 & m_2 l_2^2 + m_2 l_1 l_2 c_2 \\ m_2 l_2^2 + m_2 l_1 l_2 c_2 & m_2 l_2^2 \end{bmatrix}$$

$\boldsymbol{B}, \boldsymbol{C}, \boldsymbol{F}$ and $\boldsymbol{G}$ represent the Coriolis, centrifugal, viscous friction and gravity terms respectively:

$$\boldsymbol{B} = \begin{bmatrix} -m_2 l_1 l_2 s_2 \\ 0 \end{bmatrix}; \quad \boldsymbol{C} = \begin{bmatrix} 0 & -m_2 l_1 l_2 s_2 \\ m_2 l_1 l_2 s_2 & 0 \end{bmatrix}$$

$$\boldsymbol{F} = \begin{bmatrix} f_1 & 0 \\ 0 & f_2 \end{bmatrix}; \quad \boldsymbol{G} = \begin{bmatrix} m_2 l_2 g c_{12} + (m_1 + m_2) l_1 g c_1 \\ m_2 l_2 g c_{12} \end{bmatrix}$$

Here, the terms c_i, s_i and c_{ij} are given by $\cos\theta_i$, $\sin\theta_i$ and $\cos(\theta_i + \theta_j)$ respectively.

3.2 Design of Open-loop Controller

As proposed in Section 2, the open loop controller can be constructed by applying a periodic torque ($\tau_1 = A + B\cos\omega t$) to the joint-1 and the position of the attractive domain will be changed according to the DC component (A), amplitude (B) and the frequency (ω). The manipulator parameters are given bellow:

$$\begin{aligned} m_1 &= 3 \text{ [kg]} \\ m_2 &= 4 \text{ [kg]} \\ l_1 &= 0.3 \text{ [m]} \\ l_2 &= 0.3 \text{ [m]} \\ f_1 &= 0.8 \text{ [kgs}^{-1}\text{]} \\ f_2 &= 0.8 \text{ [kgs}^{-1}\text{]} \\ g &= 9.81 \text{ [ms}^{-2}\text{]} \end{aligned}$$

As an example, **Table 1** gives different values for A, B and ω for constructing corresponding attractors in **Fig. 3**. In this design, we set the joint-1 torque parameters as $A = 0$, $B = 4$ and $\omega = 10$ in order to drive the system state vector $\boldsymbol{x}^T = [\theta_1,\ \dot{\theta}_1,\ \theta_2,\ \dot{\theta}_2]$ towards the equilibrium point $[-\pi/2,\ 0,\ 0,\ 0]$ forming a chaotic attractor around $[-\pi/2,\ 0,\ 0,\ 0]$.

Table 1: Different A, B and ω values for constructing different attractors

Torque	A	B	ω
T_1	−18	4	10
T_2	−15	4	10
T_3	−10	4	10
T_4	0	4	10
T_5	10	4	10

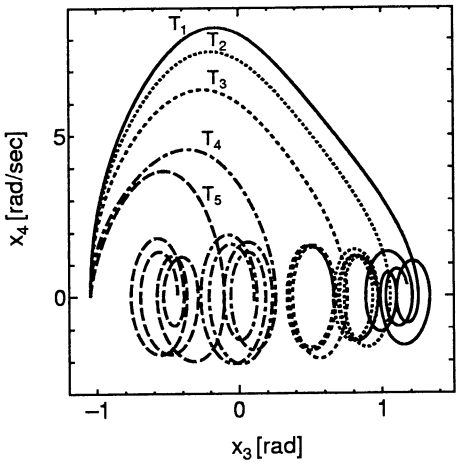


Figure 3: Construction of different attractors

Table 2: Coordinates of the fuzzy domain

P_i	x_1 [rad]	x_2 [rad/s]	x_3 [rad]	x_4 [rad/s]
P_1	−1.6708	−0.0578	0.0993	−0.0701
P_2	−1.4709	−0.0578	−0.0001	0.0944
P_3	−1.5707	0.1155	−0.0974	0.1154
P_4	−1.6559	−0.0875	−0.0999	−0.0577
P_5	−1.4747	0.0917	0.1000	−0.0577

3.3 Design of Closed-loop Controller

In this case, we construct the convex domain around the equilibrium point $[-\pi/2,\ 0,\ 0,\ 0]$ as proposed in [9]. Gain scheduling of the closed-loop controller can be carried out using the LMIs [2]. Fuzzy attractive domain is constructed with five off-equilibrium points P_i, $(i = 1, 2, .., 5)$ and the points are given in **Table 2**. Rest of the design procedure of the closed-loop controller is similar to [9]. **Figure 4** shows the resulting phase trajectories of the controlled system starting from $[-2\pi/5,\ 0,\ -\pi/3,\ 0]$. Furthermore, the torque inputs are given in **Fig. 5**.

4 Conclusions

Generated chaotic features have been used to drive the system states of the manipulator to a predefined domain using an OLC. Once it reaches to the predefined fuzzy attractive domain, a fuzzy model-based controller is employed under state feedback control to achieve reference target. By employing a periodic input, open-loop controller is constructed to drive the system towards the fuzzy attractive domain. Once it reaches to the fuzzy domain, a fuzzy model-based controller drives the system state towards the equilibrium point. The simulation result has shown the good control performance of the proposed controller.

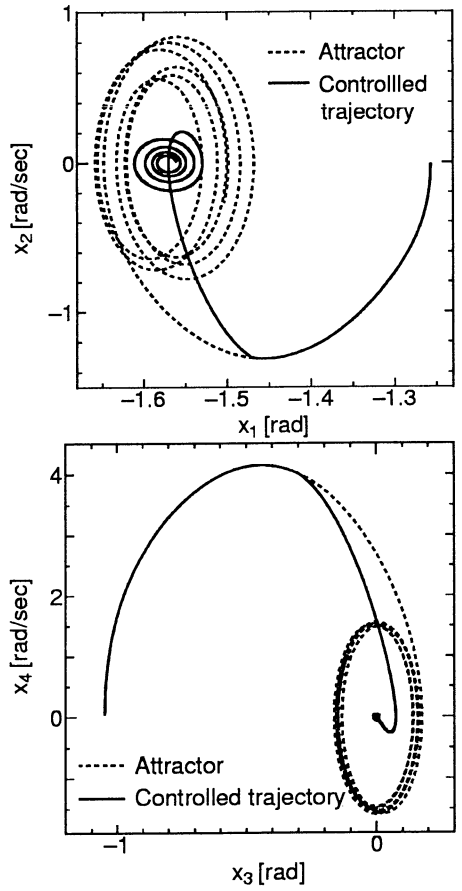


Figure 4: Controlled phase trajectory

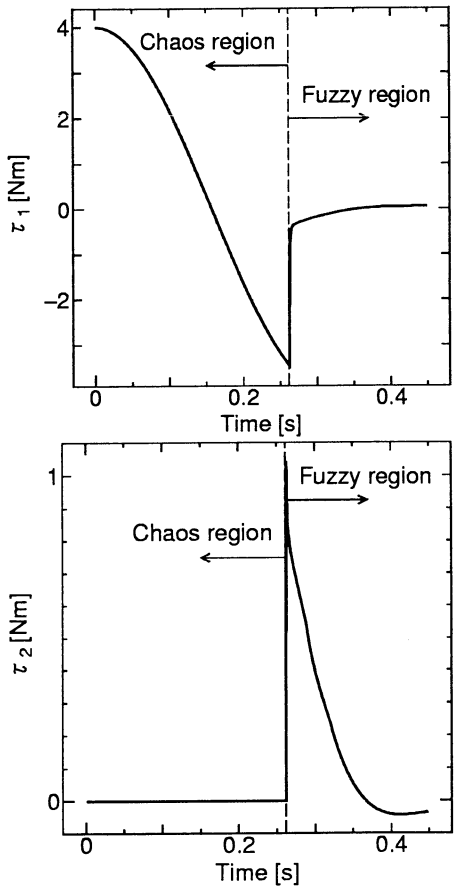


Figure 5: Torque inputs

Thus, the proposed methodology is useful for controlling nonholonomic robot manipulators.

References

[1] T. L. Vincent, "Control Using Chaos," *IEEE Control Systems Society,* vol. 17, no. 6, pp. 65–76, 1997.

[2] K. Tanaka, T. Ikeda and H. O. Wang, "Fuzzy Regulators and Fuzzy Observers: Relaxed Stability Condition and LMI-Based Designs," *IEEE Trans. on Fuzzy Systems,* vol. 6, no. 2, pp. 250–265, 1998.

[3] S. Limanond and J. Si, "Neural-Network Based Control Design: An LMI Approach," *IEEE Trans. on Neural Netwirks,* vol. 9, no. 6, pp. 1422–1429, 1998.

[4] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in Systems and Control Theory,* Philadelphia, PA: SIAM, 1994.

[5] R. Palm and D. Driankov, *Model Based Fuzzy Control,* Berlin: Springer-Verlag, 1997.

[6] T. L. Vincent and W. J. Grantham, *Nonlinear and Optimal Control Systems,* New York, NY: Wiley, 1997.

[7] H. Ying, "Constructing Nonlinear Variable Gain Controllers via the Takagi-Sugeno Fuzzy Control," *IEEE Trans. on Fuzzy Systems,* vol. 6, no. 2 pp. 226–234, 1998.

[8] T. Suzuki and Y. Nakamura, "Control of Manipulators with Free—Joints via Averaging Method," in *Procs. of IEEE Int. Conf. on Robotics and Automation (ICRA'97),* 1997, pp. 2998–3005.

[9] K. Watanabe, L. Udawatta, K. Kiguchi and K. Izumi, "Control of Chaotic systems Using Fuzzy Model-Based Regulators," Lecture Notes in Artificial Intelligence, Berlin: Springer-Verlag, 1999, pp. 248-256.

[10] Y. Nakamura, T. Suzuki and M. Koinuma, "Nonlinear Behavior and Control of a Nonholonomic Free-Joint Manipulator," *IEEE Trans. on Robotics and Control,* vol. 13, no. 6, pp. 853–862, 1997.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A Dynamic Control of Holonomic and Omnidirectional Mobile Robot Platforms with Two Active Dual-Wheel Caster Assemblies

Fuhua Han*, Keigo Watanabe**, Kazuo Kiguchi**, 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,
Saga University, 1-Honjomachi, Saga 840-8502, Japan
†E-mail : watanabe@me.saga-u.ac.jp

Abstract

In past studies, we proposed holonomic and omnidirectional mobile robots with two or more active dual-wheel caster assemblies, and derived their kinematic models. In this study, we derive their dynamic models, and implement a feedback control. The effectiveness of the proposed method is illustrated by computer simulation.

Keywords: Holonomic and omnidirectional mobile robot, Active dual-wheel caster, Kinematic model, Dynamic model, Feedback control

1 Introduction

Recently, studies on holonomic and omnidirectional mobile robots with high mobility and their control have been performed actively. However, there still remain some common problems for such a kind of robot, such as lack of surmounting stairs ability, lack of bearing load ability, their complexity, etc. In the past studies, we proposed the active dual-wheel caster assembly mechanism which can independently generate two-degrees of freedom motion. By using two or more such assemblies, a holonomic and omnidirectional mobile robot was realized. The kinematic models of the assembly and mobile robot were derived, and the effectiveness had been illustrated by some computer simulations and experiments.

On the other hand, the dynamic model is very important for the studies of its performance and its simulator. Up to now, because of its complexity, the study based on its dynamic model for the holonomic and omnidirectional mobile robot is very few. In this study, we derived the dynamic model of the holonomic and omnidirectional mobile robot using the Newton-Euler method. In order to measure the physical parameters of the mobile robot, some laboratory experiments were carried out. However, there still exist some uncertainties in this model or their parameters.

In the following sections, the dynamic model of the holonomic and omnidirectional mobile robot, control system and simulation results will be introduced.

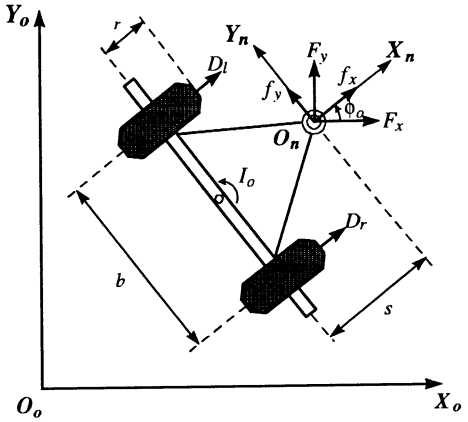


Figure 1: Model of active dual-wheel caster

2 Dynamic Model of the Active Dual-wheel Caster

Now we consider the dynamic model of an active dual-wheel caster shown in **Fig. 1**. It is assumed that the absolute coordinate system $O_o - X_o Y_o$ as a movement space of dual-wheel caster is fixed on the plane and the moving coordinate system $O_n - X_n Y_n$ is fixed on the steering center of dual-wheels caster. Here, the X_n−coordinate is set such that it has the same direction as the front of axle. Let ϕ_o denote the angle between X_o− and X_n−coordinates, f_x and f_y are the forward force and the lateral force in the moving coordinate generated from the caster, F_x and F_y are the forces generated in the absolute coordinate system. Then, we have

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} \cos\phi_o & -\sin\phi_o \\ \sin\phi_o & \cos\phi_o \end{bmatrix} \begin{bmatrix} f_x \\ f_y \end{bmatrix}. \quad (1)$$

The dynamical property for the dual-wheels caster is considered by

$$\begin{aligned} f_x &= D_l + D_r - m\dot{v}_x \\ f_y &= \tfrac{b}{2s}(D_r - D_l) - \tfrac{I_o}{s}\ddot{\phi}_o \end{aligned} \quad (2)$$

where m is the mass of the dual-wheels caster; D_l and D_r are the driving forces generated from the left- and right-

wheel, respectively; I_o is the moment of inertia around the center of rotational axis of the two wheels; s is the offset length between the axle and steering axis; and b is the distance between the two wheels (i.e., tread).

The driving system property for each wheel is assumed to be given by

$$\begin{aligned} I_w\dot{\omega}_i + c\omega_i &= ku_i - rD_i, \\ i &= l, \quad r \end{aligned} \tag{3}$$

where I_w is the moment of inertia of the wheel around the rotational axis; ω_i is the rotational rate of the left- or right-wheel toward the direction of X_n; c is the viscous friction factor for the wheel; k is the driving gain factor; u_i is the driving input torque; and r is the radius of the wheel.

3 Dynamic Model of the Mobile Robot

Now, we consider the dynamic model of an omnidirectional mobile robot with two active dual-wheel casters. Let the mobile robot move on the work space. It is assumed that the absolute coordinate system $O-XY$ is fixed on the plane and the moving coordinate system $O_m - X_mY_m$ is fixed on the center of gravity (c.g.) of the robot, as shown in **Fig. 2**. Here, X_m−coordinate is set such that it has the same direction as the forward of the robot. Let the coordinate of each dual-wheel caster be defined as $O_i - X_iY_i$, $i = a, b$. ϕ denotes the angle between $X-$ and X_m−coordinates (i.e., azimuth of the robot). F_{ix} and F_{iy} denote the forces generated from the dual-wheel caster i in the absolute coordinate system. Then the dynamical property of the mobile robot can be given by

$$\begin{aligned} M\ddot{x} &= F_{ax} + F_{bx} \\ M\ddot{y} &= F_{ay} + F_{by} \\ I\ddot{\phi} &= \tfrac{L\cos\phi}{2}(F_{ay} - F_{by}) - \tfrac{L\sin\phi}{2}(F_{ax} - F_{bx}) \end{aligned} \tag{4}$$

where M is the mass of the robot platform; I is the moment of inertia around the c.g. of the robot; L is the distance between the steering axis of the two dual-wheel casters a and b. It is assumed that the two dual-wheel casters have the same physical parameters.

If we define the state variable vector for the mobile robot as $\boldsymbol{x} = [x\ y\ \phi\ \dot{x}\ \dot{y}\ \dot{\phi}]^T$, the input variable vector as $\boldsymbol{u} = [u_{al}\ u_{ar}\ u_{bl}\ u_{br}]^T$ and the output variables as $\boldsymbol{y} = [\dot{x}\ \dot{y}\ \dot{\phi}]^T$, from Eq. (1) to Eq. (4), we can obtain

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \\ \boldsymbol{y} &= C\boldsymbol{x} \end{aligned} \tag{5}$$

where

$$C = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$A = \begin{bmatrix} O_{3\times3} & I_{3\times3} \\ O_{3\times3} & A_o^{-1}B_o \end{bmatrix}$$

and

$$B = \begin{bmatrix} O_{3\times4} \\ A_o^{-1}C_o \end{bmatrix}.$$

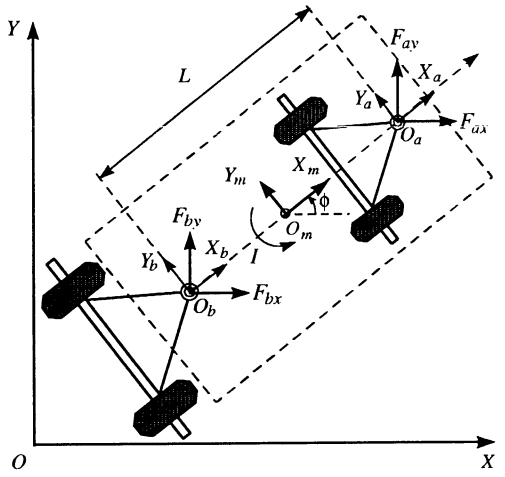


Figure 2: Model of omnidirectional mobile robot

If we define the following representations

$$A_o = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

$$B_o = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix}$$

$$C_o = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \end{bmatrix}$$

it follows that

$$\begin{aligned} A_{11} &= M + (P_1 - P_2)(\sin^2\phi_a + \sin^2\phi_b) + 2P_2 \\ A_{12} &= \tfrac{P_2-P_1}{2}(\sin 2\phi_a + \sin 2\phi_b) \\ A_{13} &= \tfrac{L}{2}(P_2 - P_1)[\sin\phi_a\cos(\phi-\phi_a) - \sin\phi_b\cos(\phi-\phi_b)] \\ A_{21} &= \tfrac{P_2-P_1}{2}(\sin 2\phi_a + \sin 2\phi_b) \\ A_{22} &= M + (P_1 - P_2)(\cos^2\phi_a + \cos^2\phi_b) + 2P_2 \\ A_{23} &= \tfrac{L}{2}(P_1 - P_2)[\cos\phi_a\cos(\phi-\phi_a) - \cos\phi_b\cos(\phi-\phi_b)] \\ A_{31} &= (P_1 - P_2)[\sin(\phi-\phi_a)\cos\phi_a - \sin(\phi-\phi_b)\cos\phi_b] \\ A_{32} &= (P_1 - P_2)[\sin(\phi-\phi_a)\sin\phi_a - \sin(\phi-\phi_b)\sin\phi_b] \\ A_{33} &= \tfrac{L}{2}(P_2 - P_1)[\sin^2(\phi-\phi_a) - \sin^2(\phi-\phi_b)] - P_1L + \tfrac{2I}{L} \end{aligned}$$

$$\begin{aligned} B_{11} &= \tfrac{P_2-P_1}{2}(\dot{\phi}_a\sin 2\phi_a + \dot{\phi}_b\sin 2\phi_b) - 2P_4 \\ &\quad -(P_3 - P_4)(\sin^2\phi_a + \sin^2\phi_b) \\ B_{12} &= (P_2 - P_1)(\dot{\phi}_a\sin^2\phi_a + \dot{\phi}_b\sin^2\phi_b) - P_2(\dot{\phi}_a + \dot{\phi}_b) \\ &\quad + \frac{P_3 - P_4}{2}(\sin 2\phi_a + \sin 2\phi_b) \\ B_{13} &= \tfrac{L}{2}(P_2 - P_1)[(\dot{\phi} - \dot{\phi}_a)\sin\phi_a\sin(\phi-\phi_a) \\ &\quad +(\dot{\phi} - \dot{\phi}_b)\sin\phi_b\sin(\phi-\phi_b)] - \frac{L}{2}(\dot{\phi}_a - \dot{\phi}_b)P_2\cos\phi \end{aligned}$$

$$+\frac{L}{2}(P_3-P_4)[\sin\phi_a\cos(\phi-\phi_a)-\sin\phi_b\cos(\phi-\phi_b)]$$

$$B_{21}=(P_1-P_2)(\dot{\phi}_a\cos^2\phi_a+\dot{\phi}_b\cos^2\phi_b)+P_2(\dot{\phi}_a+\dot{\phi}_b)$$
$$+\frac{P_3-P_4}{2}(\sin 2\phi_a+\sin 2\phi_b)$$

$$B_{22}=\frac{P_1-P_2}{2}(\dot{\phi}_a\sin 2\phi_a+\dot{\phi}_b\sin 2\phi_b)-2P_4$$
$$-(P_3-P_4)(\cos^2\phi_a+\cos^2\phi_b)$$

$$B_{23}=\frac{L}{2}(P_1-P_2)[(\dot{\phi}-\dot{\phi}_a)\cos\phi_a\sin(\phi-\phi_a)$$
$$-(\dot{\phi}-\dot{\phi}_b)\cos\phi_b\sin(\phi-\phi_b)]-\frac{L}{2}(\dot{\phi}_a-\dot{\phi}_b)P_2\sin\phi$$
$$-\frac{L}{2}(P_3-P_4)[\cos\phi_a\cos(\phi-\phi_a)-\cos\phi_b\cos(\phi-\phi_b)]$$

$$B_{31}=(P_1-P_2)[\dot{\phi}_a\cos\phi_a\cos(\phi-\phi_a)$$
$$-\dot{\phi}_b\cos\phi_b\cos(\phi-\phi_b)]+(\dot{\phi}_a-\dot{\phi}_b)P_2\cos\phi$$
$$+(P_3-P_4)[\sin\phi_a\cos(\phi-\phi_a)-\sin\phi_b\cos(\phi-\phi_b)]$$

$$B_{32}=(P_1-P_2)[\dot{\phi}_a\sin\phi_a\cos(\phi-\phi_a)$$
$$-\dot{\phi}_b\sin\phi_b\cos(\phi-\phi_b)]+(\dot{\phi}_a-\dot{\phi}_b)P_2\sin\phi$$
$$-(P_3-P_4)[\cos\phi_a\cos(\phi-\phi_a)-\cos\phi_b\cos(\phi-\phi_b)]$$

$$B_{33}=\frac{L(P_1-P_2)}{4}[(\dot{\phi}-\dot{\phi}_a)\sin 2(\phi-\phi_a)-P_4L$$
$$+(\dot{\phi}-\dot{\phi}_b)\sin 2(\phi-\phi_b)]$$
$$-\frac{L}{2}(P_3-P_4)[\cos^2(\phi-\phi_a)+\cos^2(\phi-\phi_b)].$$

Furthermore, we have

$$C_{11}=P_5\cos\phi_a+P_6\sin\phi_a$$
$$C_{12}=P_5\cos\phi_a-P_6\sin\phi_a$$
$$C_{13}=P_5\cos\phi_b+P_6\sin\phi_b$$
$$C_{14}=P_5\cos\phi_b-P_6\sin\phi_b$$
$$C_{21}=P_5\sin\phi_a-P_6\cos\phi_a$$
$$C_{22}=P_5\sin\phi_a+P_6\cos\phi_a$$
$$C_{23}=P_5\sin\phi_b-P_6\cos\phi_b$$
$$C_{24}=P_5\sin\phi_b+P_6\cos\phi_b$$
$$C_{31}=P_5\sin(\phi_a-\phi)-P_6\cos(\phi_a-\phi)$$
$$C_{32}=P_5\sin(\phi_a-\phi)+P_6\cos(\phi_a-\phi)$$
$$C_{33}=P_5\sin(\phi-\phi_b)+P_6\cos(\phi-\phi_b)$$
$$C_{34}=P_5\sin(\phi-\phi_b)-P_6\cos(\phi-\phi_b)$$

where

$$P_1=\frac{b^2I_w}{2r^2s^2}+\frac{I_o}{s^2},\quad P_2=\frac{2I_w}{r^2}+m$$
$$P_3=\frac{b^2c}{2r^2s^2},\quad P_4=\frac{2c}{r^2}$$
$$P_5=\frac{k}{r},\quad P_6=\frac{bk}{2rs}.$$

Additionally, it should be noted that $\dot{\phi}_a$ and $\dot{\phi}_b$ are given by

$$\begin{aligned}\dot{\phi}_a &= \tfrac{1}{s}\dot{y}\cos\phi_a-\tfrac{1}{s}\dot{x}\sin\phi_a+\tfrac{L}{2s}\dot{\phi}\cos(\phi-\phi_a)\\ \dot{\phi}_b &= \tfrac{1}{s}\dot{y}\cos\phi_b-\tfrac{1}{s}\dot{x}\sin\phi_b-\tfrac{L}{2s}\dot{\phi}\cos(\phi-\phi_b).\end{aligned}\quad(6)$$

4 Feedback Control System

In order to realize a feedback control to the mobile robot, the input vector is defined by the following relation:

$$\boldsymbol{u}=J\boldsymbol{u}_w \quad(7)$$

where, $\boldsymbol{u}_w=[u_{\dot{x}}\ u_{\dot{y}}\ u_{\phi}]^T$ is the input vector in the absolute coordinate system, and J is the Jacobian matrix given by

$$J=\begin{bmatrix}J_{11}&J_{12}&J_{13}\\J_{21}&J_{22}&J_{23}\\J_{31}&J_{32}&J_{33}\\J_{41}&J_{42}&J_{43}\end{bmatrix}$$

in which

$$J_{11}=\frac{1}{4}\cos\phi_a+\frac{s}{2b}\sin\phi_a$$
$$J_{12}=\frac{1}{4}\sin\phi_a-\frac{s}{2b}\cos\phi_a$$
$$J_{13}=\frac{1}{2L}\sin(\phi_a-\phi)-\frac{s}{Lb}\cos(\phi_a-\phi)$$
$$J_{21}=\frac{1}{4}\cos\phi_a-\frac{s}{2b}\sin\phi_a$$
$$J_{22}=\frac{1}{4}\sin\phi_a+\frac{s}{2b}\cos\phi_a$$
$$J_{23}=\frac{1}{2L}\sin(\phi_a-\phi)+\frac{s}{Lb}\cos(\phi_a-\phi)$$
$$J_{31}=\frac{1}{4}\cos\phi_b+\frac{s}{2b}\sin\phi_b$$
$$J_{32}=\frac{1}{4}\sin\phi_b-\frac{s}{2b}\cos\phi_b$$
$$J_{33}=\frac{1}{2L}\sin(\phi-\phi_b)+\frac{s}{Lb}\cos(\phi-\phi_b)$$
$$J_{41}=\frac{1}{4}\cos\phi_b-\frac{s}{2b}\sin\phi_b$$
$$J_{42}=\frac{1}{4}\sin\phi_b+\frac{s}{2b}\cos\phi_b$$
$$J_{43}=\frac{1}{2L}\sin(\phi-\phi_b)-\frac{s}{Lb}\cos(\phi-\phi_b).$$

The output error vector is defined by

$$\boldsymbol{e}=\boldsymbol{y}_r-\boldsymbol{y} \quad(8)$$

where $\boldsymbol{y}_r=[\dot{x}_r\ \dot{y}_r\ \phi_r]^T$ is the reference vector. Here we adopt P-servo controller to control the rotational angle of

the robot, and PI-servo controller to control each velocity. On controlling azimuth of the robot, it can be found that if the P gain is larger, overshot can be caused; and that if the k_p is smaller, rise time is very slow. So we adopt a fuzzy P-servo controller with one input to the antecedent part. Membership function is shown in **Fig. 3**.

The fuzzy estimated gain $\hat{k_p}$ is obtained from the fuzzy reasoning output given by

$$\hat{k_p} = \sum_{i=1}^{n} \mu_i \bar{k}_i \tag{9}$$

where n is the number of fuzzy rules; $\bar{k}_i$ is the consequent values of the rule i; μ_i denotes the membership values of fuzzy set i.

5 Simulation Results

The simulation was performed to verify the effectiveness of the dynamic model and its controller. Simulation time was 20 [s]; sampling period was 15 [ms]; the initial value of the state variable was $\boldsymbol{x} = [0\ 0\ 0\ 0\ 0\ 0]^T$, and the initial posture of each dual-wheel caster was 0 [rad]. The physical parameters of each caster and the robot platform were given by

$$M = 21.97\ [\text{kg}],\ m = 6.51\ [\text{kg}]$$

$$I = 3.336\ [\text{kg m}^2],\ I_o = 0.117\ [\text{kg m}^2]$$

$$I_w = 0.00076\ [\text{kg m}^2],\ c = 1.5\ [\text{kg m}^2/\text{ s}]$$

$$r = 0.05\ [\text{m}],\ s = 0.075\ [\text{m}]$$

$$b = 0.26\ [\text{m}],\ L = 0.8\ [\text{m}]$$

$$k = 1.0$$

Here, PI-gains for each velocity were assumed to be

$$k_{p\dot{x}} = 20,\ k_{i\dot{x}} = 20,\ \ k_{p\dot{y}} = 10,\ k_{i\dot{y}} = 30.$$

The reference vector was $\boldsymbol{y}_r = [3\sqrt{2}/20\ \ 3\sqrt{2}/20\ \ \pi/4]^T$. Control results are shown in **Fig. 4**.

6 Conclusions

In this paper, we have derived the dynamic model of an omnidirectional mobile robot with two active dual-wheel casters. It can be easily found that this is a nonlinear system with respect to its azimuth. In order to realize a feedback control to the mobile robot, we derived Jacobian matrix, adopted P-servo controller or fuzzy P-servo controller to control the rotational angle of the robot, and PI-servo controller to control each velocity. The effectiveness was illustrated by computer simulation.

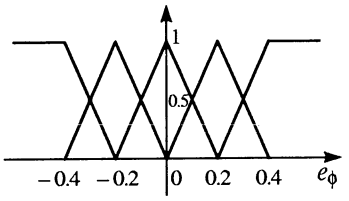


Figure 3: Membership function

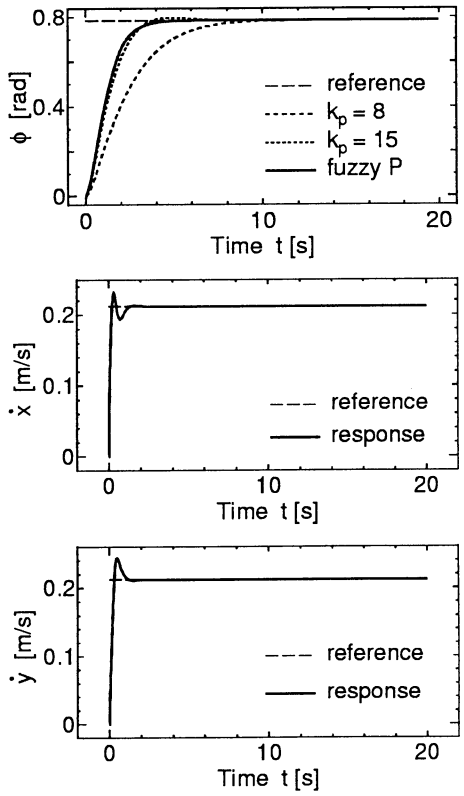


Figure 4: Control results of rotational angle, X-directional velocity and Y-directional velocity

References

[1] F. Han, K. Watanabe, and K. Izumi, "Kinematic Analysis for an Omnidirectional Mobile Robot with Active Dual-Wheel Casters," in *Procs. of the 16th Annual Conf. of the Robotics Society of Japan,* 1998, pp. 205–206.

[2] K. Watanabe, Y. Shiraishi, and J. Tang, "Construction of the Dynamic Model for an Omnidirectional Mobile Robot," in *Procs. of the 14th SICE Kyushu Branch Annual Conf.,* 1995, pp. 269–272.

[3] K. Watanabe, "Control of an Omnidirectional Mobile Robot," in *Procs. of the 2nd Int. Conf. on Knowledge-based Intelligent Electronic Systems (KES '98), Adelaide, Australia,* 1998, vol. 1, pp. 51–60.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Environment Cognition for a Quadruped Robot Using an ART-Based Neural Network

Xuedong Chen*, Keigo Watanabe**, Kazuo Kiguchi** and Kiyotaka Izumi***
*Graduate School of Science and Engineering,
Faculty of Engineering Systems and Technology,
**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
†E-mail : watanabe@me.saga-u.ac.jp

Abstract

To implement adaptive navigation for a quadruped robot, one of the most important and fundamental tasks is cognition of the environment category of the robot. An attempt to formulate the environment cognition problem is made in an identification framework that exploits the computational features of an ART-based neural network. The ART-II model is chosen with the hope that the network would learn a single presentation of an input pattern and that it would be self organizing, with the clustering of input patterns being done without an explicit specification of the classification desired. The inputs of the proposed environment recognizer is composed of the parameter of environment limits and the obstacle-distances between the robot and the obstacles within the environment range. Therefore, the recognizer can adapt itself to any scope of environment by adjusting the input of the environment limits. The effectiveness of the proposed approach is demonstrated through some examples.

Keywords:Quadruped robot, Environment cognition, Adaptive resonance theory, ART-based neural network.

1 Introduction

It is known that a quadruped robot can realize walking on the rough ground with the help of the appropriate control of gait. If there is not the navigation control, however, the walking is only at locomotion level, i.e., at a rather low level in the overall behavior control for the quadruped robot, because the robot has to not only adapt itself to the rough ground profile but also be able to steer itself towards the specified target and prevent itself from collision with obstacles. To implement the adaptive navigation for a quadruped robot in a cluttered environment, one of the most important and fundamental tasks is cognition of the environment category of the robot.

Since the Adaptive Resonance Theory (ART) [1–8] is powerful in self-organization of stable category recognition, we here present an attempt to formulate the environment identification problem, for the purpose of adaptive control, in an identification network that exploits the computational features of an ART-based neural network. The ART-II model is chosen with the hope that the network would learn a single presentation of an input pattern and that it would be self organizing, with the clustering of input patterns being done without an explicit specification of the classification desired. The additional advantages of using the ART-II that we expected were: 1) elimination of the need to retrain the network when there is a change in the prototype set of input patterns or when more patterns are added to it; 2) control of the classification in terms of fine and coarse categorization via the attentional vigilance parameter of the network; and 3) the network's capability of unsupervised learning.

In this paper, the environment category of the quadruped robot is presented on the basis of selecting the sensor information and the environment limits. The cognition of the environment category is implemented in the form of the ART-II neural network [2] in the supervised learning mode. The principle of the ART-based environment recognizer is also described.

2 Environment Category

The environment of a quadruped robot is defined as the area surrounding the robot with the radius R centered at the robot. The radius R is an important parameter for the environment cognition because the environment scope should be changeable for the different control requirements by adjusting R. On the other hand, categorizing the environment depends on the exploration and measurement of the obstacles in the environment by vision system. Here, we assume the distances between the quadruped robot and the obstacles are obtainable by using some sensors. Due to the different requirements of environment cognition, the obstacle-distances can be properly selected. For instance, we can adopt the obstacle-distances in the front direction, the left direction and the right direction as the parameters for the environment cognition, moreover, we can also use the obstacle-distances in these directions adding the

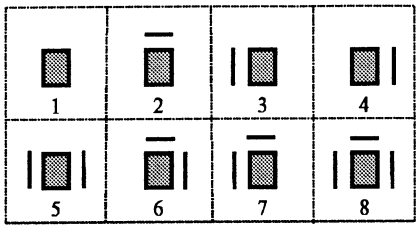


Figure 1: Categories of environment of a quadruped robot for the case of three obstacle-distances

left-front direction and the right-front direction as the parameters for the environment cognition, etc. It should be noted that the fineness of cognition is in direct proportion to the number of obstacle-distances. Therefore, the appropriate environment cognition can be realized by selecting the corresponding environment scope and the obstacle-distances.

The obstacle-distances in the front direction, the left direction, the right direction, the left-front direction and the right-front direction of the quadruped robot are denoted by d_f, d_l, d_r, d_{lf} and d_{rf}, respectively. Clearly, the environment can be considered to be the non-obstacle one as long as all of the obstacle-distances are larger than R, otherwise the environment is the obstacle-environment. Based on the fact, there are eight possible categories of environment for the case of d_f, d_l and d_r selected as the obstacle-distances parameters shown in **Fig. 1**, whereas there are thirty-two possible categories of environment for the case of d_f, d_l, d_r, d_{lf} and d_{rf} adopted as the distance parameters shown in **Fig. 2**. An ART-based neural network is introduced for the cognition of environment category according to the vision information.

3 Environment Recognizer

The environment cognition is implemented in the form of the ART-II neural network [2]. The ART-II is a member of the class of Adaptive Resonance architectures that is designed to handle both binary and analog patterns and is a modification over the first proposed ART architecture called ART-I [1]. **Figure 3** depicts the ART-II network topology. This network includes the principal components of all ART modules, namely an attentional subsystem, which contains an input representation field F_1 and a category representation field F_2, and an orienting subsystem, which interacts with the attentional subsystem to carry out an internally controlled search process. The two fields are linked by both a bottom-up $F_1 \rightarrow F_2$ adaptive filter and a top-down $F_2 \rightarrow F_1$ adaptive filter. A path from the ith F_1 node to the jth F_2 node contains a long term memory (LTM) trace, or adaptive weight z_{ij}, a path from the jth F_2 node to the ith F_1 node contains a weight z_{ji}. These weights gate, or multiply, path signals between fields. In Fig. 3, the global network parameters and constants are as follows: I_i ($i = 1, \cdots, M$) is the input of the ith channel of F_1, $a = 10.0$, $b = 10.0$, $c = 0.10$, $d = 0.9$, $e = 0.0$ and $\theta = 1/\sqrt{M}$. The local STM activities p_i, q_i, u_i, w_i, and x_i computed at the

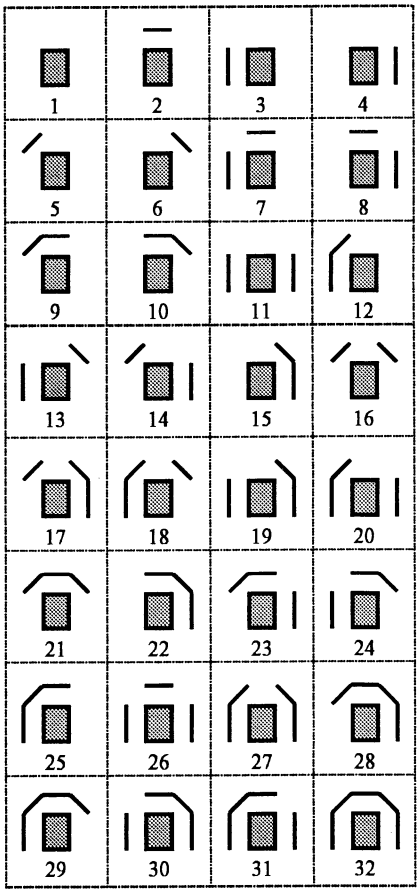


Figure 2: Categories of environment of a quadruped robot for the case of five obstacle-distances

STM stage F_1 (all equations expressed in dimensionless form) are as follows:

$$p_i = u_i + \sum_j g(y_j) z_{ji} \tag{1}$$

$$q_i = \frac{p_i}{e + |\boldsymbol{p}|} \tag{2}$$

$$u_i = \frac{v_i}{e + |\boldsymbol{v}|} \tag{3}$$

$$v_i = f(x_i) + bf(q_i) \tag{4}$$

$$w_i = I_i + au_i \tag{5}$$

$$x_i = \frac{w_i}{e + |\boldsymbol{w}|} \tag{6}$$

for $i = 1, \cdots, M$ and $j = M+1, \cdots, N$, where $|\boldsymbol{v}|$ represents the L_2 norm of the vector $\boldsymbol{v}$. In addition, the thresholding function is adopted as follows

$$f(t) = \begin{cases} 0 & \text{if } 0 \le t < \theta \\ t & \text{if } t \ge \theta. \end{cases} \tag{7}$$

The local STM activity, T_j computed at the STM stage F_2 is given by:

$$T_j = \sum_i p_i z_{ij} \tag{8}$$

for $j = M+1, \cdots, N$. F_2 makes a choice if Jth F_2 node becomes maximally active, while all other nodes are inhibited, when

$$T_J = \max\{T_j : j = M+1, \cdots, N\}. \tag{9}$$

The main elements of the gated dipole field dynamics may be characterized as follows: if T_J=max$\{T_j$: the jth F_2 node hasn't been reset on current trial$\}$, then $g(y_J) = d$, otherwise $g(y_J) = 0$.

The top-down and bottom-up LTM trace equations are as follows:

$$F_2 \to F_1 : \frac{d}{dt} z_{ji} = g(y_j)[p_i - z_{ji}], \tag{10}$$

$$F_1 \to F_2 : \frac{d}{dt} z_{ij} = g(y_j)[p_i - z_{ij}]. \tag{11}$$

When F_2 makes a choice and the Jth F_2 node is selected during contrast enhancement process (winner takes all or on center off surround selection procedure), Eqs. (31) and (32) can be modified as:

$$\frac{d}{dt} z_{Ji} = d[p_i - z_{Ji}] = d(1-d)[\frac{u_i}{1-d} - z_{Ji}],, \tag{12}$$

$$\frac{d}{dt} z_{iJ} = d[p_i - z_{iJ}] = d(1-d)[\frac{u_i}{1-d} - z_{iJ}], \tag{13}$$

for all $j \neq J$, $dz_{ji}/dt = 0$, and $dz_{ij}/dt = 0$.

The orienting subsystem consisting of the reset mechanism resets F_2 whenever the input pattern is active and

$$\frac{\rho}{e + |\boldsymbol{r}|} > 1 \tag{14}$$

where,

$$r_i = \frac{u_i + cp_i}{e + |\boldsymbol{w}| + |\boldsymbol{cp}|}. \tag{15}$$

The network initialization involves setting all the network parameters at typical values indicated in the section on network equations. The top-down connection weights are all initially set equal to 0. The bottom-up connection weights are initialized at:

$$z_{ij} = \frac{1}{2(1-d)\sqrt{M}}. \tag{16}$$

The ART-II network is implemented on a digital computer. The network is used in the supervised learning mode and is trained off-line before its inclusion into the control loop. The training data for the net, including the environment radius R and the typical data of the obstacle-distances, are obtained from the real environment categorizing. Note that, if the obstacle-distance is larger than R, or there is no obstacle in the corresponding direction, then this obstacle-distance is equal to R.

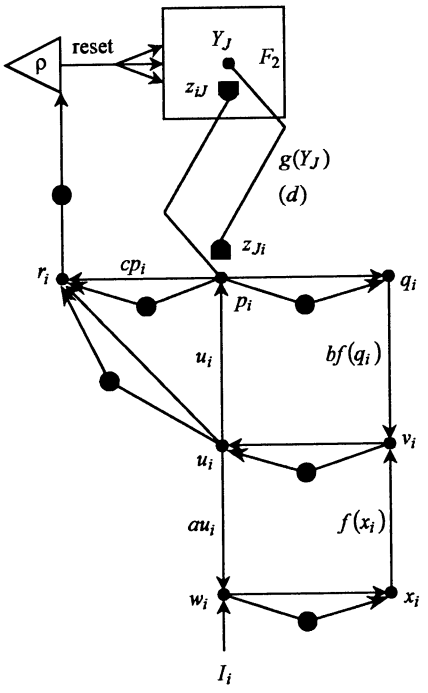


Figure 3: The ART-II network topology

Once the training data are available, the network is configured so that the number of nodes in the feature representation field F_1, which corresponds to the dimension of the input feature vector composed of the environment radius and the selected obstacle-distances in present application, is fixed. The number of processing nodes in the category representation field F_2, is generally greater than the total number of input patterns in the prototype set. Each pattern in the prototype set is sequentially presented to the network once. A second cyclic presentation of the prototype set may be made for a stable category confirmation. Subsequent presentations do not alter the resulting category structure. The category structure represents the stable space partitioning of the neural network depending on the number of stable categories established during training and the different feature vectors that were classified into these categories. During training, the attentional vigilance parameter is set at its highest value (0.99) to ensure a high resolution of the resulting category structure.

When the network is presented with a feature vector for the first time, it is encoded in the long term memory (LTM) through modification of the LTM connection weights. A node is allocated in the network's category representation field F_2 to represent the pattern. The parameters associated with the feature vector now get assigned to this allocated F_2 node. On presentation of subsequent feature vectors, the network's orienting subsystem first determines closeness of match between the pattern currently imposed on the network and any of the patterns that have previously been seen. Since the

vigilance parameter is set high, a new node is allocated for the pattern. However, if the current pattern happens to be closely matched to the one that the network has already seen, it is clustered into the same category. It is therefore possible to partition the network's state space so that each partition serves as an attractor for a particular type of response characterized by its feature vector. The vigilance parameter helps to control the coarseness or fineness of classification desired. After completion of training, the top-down and the bottom-up connection weights of the network, along with the network parameters, are saved into the computer memory. The above process is repeated for different sets of values of the obstacle-distances. After its inclusion into the control loop, the trained network can realize the rapid identification of the environment category according to the desired environment scope and the vision information of the robot.

4 The Practical Example

Based on the principle of the environment cognition as above, the recognizers for the case of three obstacle-distances (d_f, d_l and d_r) (recognizer 1) and the case of five obstacle-distances (d_f, d_l, d_r, d_{lf} and d_{rf}) (recognizer 2) are designed. **Figure 4** shows the quadruped robot walking on the terrain with some obstacles, where $R_1 = 1$ [m], $R_2 = 1.5$ [m], $R_3 = 2$ [m], $d_f = 1.8$ [m], $d_l = 1.2$ [m], $d_r = 0.8$ [m], $d_{lf} = 2.25$ [m] and $d_{rf} = 1.45$ [m]. The environment category can be recognized by the proposed recognizers. The results are as follows:

If $R = R_1$, the outputs of the recognizer 1 and the recognizer 2 are the 4th category and the 4th category, respectively;

If $R = R_2$, the outputs of the recognizer 1 and the recognizer 2 are the 5th category and the 19th category, respectively;

If $R = R_3$, the outputs of the recognizer 1 and the recognizer 2 are the 8th category and the 30th category, respectively.

The above cognition results coincide with the environment category shown in Fig. 1 and Fig. 2.

5 Conclusions

The definition of the environment and the environment category of a quadruped robot have been presented in this paper. An approach for the environment cognition has been attempted, which involves the application of a neural network architecture based on the adaptive resonance theory. It was shown that it was possible to train the ART-based network off-line (supervised learning mode) to identify the environment category based on the selection of the desired environment scope and the concerned obstacle-distances. The correct cognition of the environment makes the adaptive navigation of a quadruped robot walking in a cluttered environment feasible and efficient. The effectiveness of the proposed recognizer was demonstrated through the practical example.

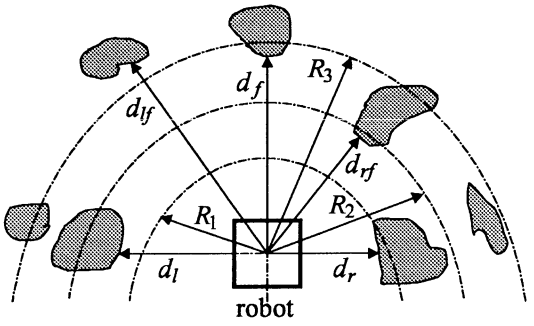


Figure 4: The practical example for the environment cognition of the quadruped robot

References

[1] G. A. Carpenter and S. Grossberg, "A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine," *Computer Vision, Graphics, and Image Processing,* vol. 37, pp. 54–115, 1987.

[2] ——, "ART 2: Self-Organization of Stable category Recognition Codes for Analog Input Patterns," *Applied Optics,* vol. 26, no. 23, pp. 4917–4930, 1987.

[3] G. A. Carpenter, S. Grossberg and D. B. Rosen, "Fuzzy ART: Fast Stable Learning and Categorization of Analog Patterns by an Adaptive Resonance System," *Neural Networks,* vol. 4, pp. 759–771, 1991.

[4] ——, "ART 2-A: An Adaptive Resonance Algorithm for Rapid Category Learning and Recognition," *Neural Networks,* vol. 4, pp. 493–504, 1991.

[5] M. Georgiopoulos, G. L. Heileman and J. X. Huang, "Properties of Learning Related to Pattern Diversity in ART 1," *Neural Networks,* vol. 4, pp. 751–757, 1991.

[6] C. J. Lin and C. T. Lin, "Reinforcement Learning For an ART-Based Fuzzy Adaptive Learning Control Network," *IEEE Trans. on Neural Networks,* vol. 7, no. 3, pp. 709–731, 1996.

[7] ——, "An ART-Based Fuzzy Adaptive Learning Control Network," *IEEE Trans. on Fuzzy Systems,* vol. 5, no. 4, pp. 477–496, 1997.

[8] A. Filippidis, L. C. Jain and P. Lozo, "Degree of Familiarity ART 2 in Knowledge-Based Landmine Detection," *IEEE Trans. on Neural Networks,* vol. 10, no. 1, pp. 186–193, 1999.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Symbiotic Artifact Design by DOILS: Design Organization by Information in Life Systems

Osamu Katai, Hiroshi Kawakami and Osamu Toba

Dept. of Systems Science, Graduate School of Informatics,
Kyoto University, Kyoto, 606-8501, JAPAN
{katai,kawakami,toba}@sys.i.kyoto-u.ac.jp

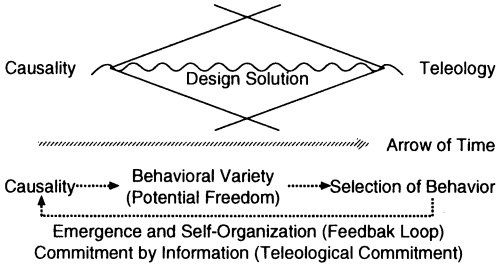
Abstract

The basic concepts and the current research activities of our project: "*System Theory of Bio-Informatics and its Extension towards Engineering Design Theory*" funded by JSPS (Japan Society for the Promotion of Science) in the RFTF (Research for the Future) Program is introduced. Particular emphases are put in our project not only on systems theoretical analyses of the roles and the effects of information on life systems but also on formalizing theories and methodologies of system synthesis that are expected to provide novel system concepts with high abilities to adapt to and to coordinate interactions with the outside environments such as shown as sophisticated robots capable of adapting to environmental changes and of high touch communications with humans.

1 Introduction

For designing symbiotic artifacts, we will consider the basic principles and methods of designing and coordinating decentralized and distributed artifact systems by referring to the roles of information in life systems through which effective collaboration and symbiosis with human beings, other artifacts and environments will harmoniously be attained. For this purpose, we will focus our attentions to the modularity of system organizations [1] and the interactions among modular subsystems. Particular attentions are payed to metalevel interrelationship between module composition and interaction process formation, mutual boundedness of modules via their interactions, the way to incorporate teleological rationality and purposeful behavior of modules, enhancement and enlargement of the boundaries of module activities, the internalization of environmental complexities into modules via interactions with the outside worlds such as learning, adaptation, memorization, internal modeling and evolution, the formation and organization of higher order relationship among modules such as the so called "symbiosis",decentralization and decomposition of composed and formed organizations, functional role formation and differentiation by referring to own and other modules, homogenuity and inhomogenuity of modules, behavioral freedom of modules, functional composition and decomposition of modules via teleological and nonteleological interactions, and temporal complexities of interactions among modules.

2 Specificity of Life System in Temporal Structures



Reduction of Teleology to Causality

- Control Theory
- Evolution
- Arrow of Time
  - — Teleology: Reverse Direction
  - — Causality: Normal Direction

Selection

- Elapse of Time

Life System

- Subjective Commitment by Information for Selecting Possible Alternatives
- Complexity of Time Structure
  - — Multiplexity of Causality
  - — Multiplexity of Time Elapse

Figure 1: Teleology and Causality, and System Design

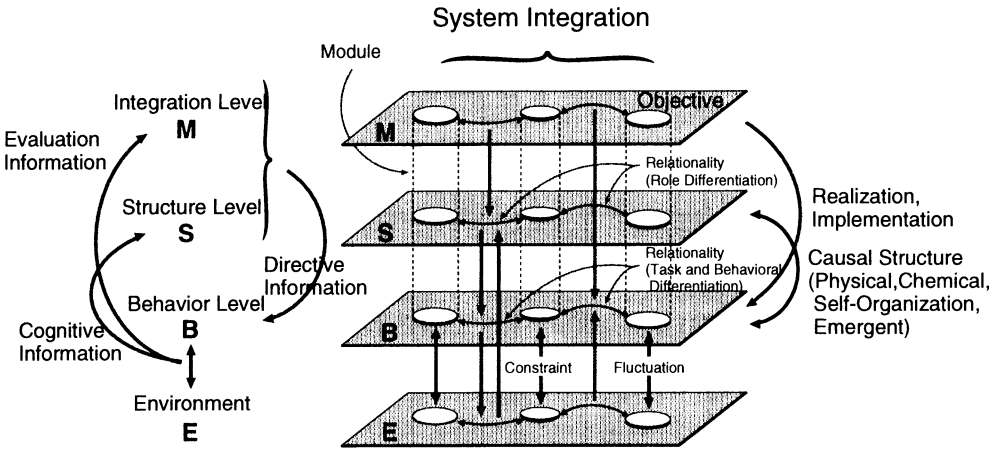


Figure 2: System Organization

Usually, as shown in Fig. 1, the "arrow of time" which naturally reflects the distinction of teleology and causality is not so simple in the case of life systems. This is because it is sometimes the case that feedback control loop or a loop of higher order formation such as feedback loop of order forming information for self-organization or order emergence are involved in the systems and hence, the arrow of time is not simply a unidirected one. Also, life systems utilizes information to select the direction of order formation or order emergence in an active manner by selecting potential varieties of their actions or their cognitive states. Thus, the life systems are of complexities in their "time structures", that is, they involve the "multiplexity of causality" and also the "multiplexity of time elapse".

3 Symbiotic Artifacts as Multi-body System of Modules

The above complexity of translation may be relaxed by introducing hierarchical modular structures composed of three layers, i.e., Integration Layer **M**, Structural Layer **S** and Behavioral Layer **B**. As shown in Table 1, we grasp on layer M the values, objectives and functions or roles of a module, and the collaboration or conflict among the modules can be analyzed. On layer S, constraint, boundary condition and allowance of behavioral variety are grasped. Also, the behavioral layer B represents the event occurrence structure along with the elapse of time.

On this layered structure, the causal effects can be interpreted as the effect of S (structure) on B (behavior), and the self-organization, emergence and autopoiesis phenomena can be seen as the effect of B on S.

Moreover, T. Yoshida [2] introduced these basic "role of information" in his study on information theory of self-organizing systems. Namely, as shown in Table 1, Cognition, Direction and Evaluation are the main roles, and they respectively are interpreted as the function of linking interactions with environment with S, linking A and M with B, and linking B and interactions with environment with M.

Finally, we come to the system structure (system organization) which can be regarded as a **"symbiotic multi-body system of modules"** as shown in Fig. 2, where the notion of symbiosis introduced by K. Kurokawa [3] is examined by a research committee in Keidanren, which is summarized as following:

> **The Definition of Symbiosis**: *"Generation and development of more than one entities with collaboration yielding an order formation in higher, wider and pluralistic values and dimensions"* [3]

4 Dual Directedness of System Formation

Our system design concept is specific, as shown in Table 2, in the sense that even thought the usual real living organisms have unidirectional tendency towards

Table 1: System Concept: **"Symbiotic Multi-body System of Modules"**

Three Aspects of Modules	
Integration (M)	Value, Objective, Function·Role, Collaboration·Conflict
Structure (S)	Constraint·Boundary, Condition, Allowance of Behavioral Variety (Freedom)
Behavior (B)	Event Structure along with the Arrow of Time

Structure → Behavior:	Causal Effect
Behavior → Structure:	Self-Organization, Emergence, Autopoiesis

Role of Information (Commitment)		
Cognition:	Interaction with Environment	→ S
Direction:	S, M	→ B
Evaluation:	B, Interaction with Environment	→ M

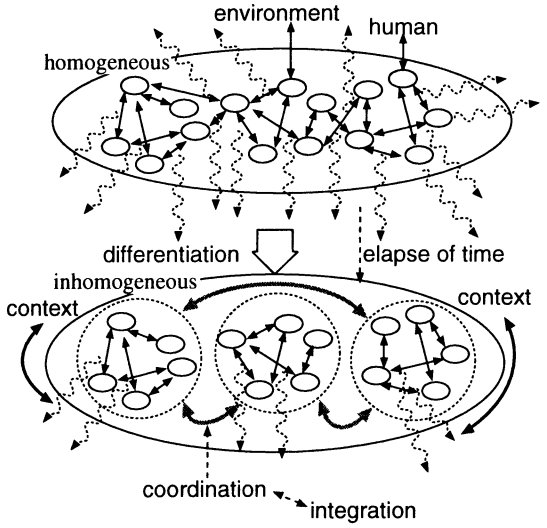


Figure 3: Differentiation and integration of interactions and modules

forming an order. That is, they never go in the opposite direction, i.e., going towards disorganization. But our systems are permitted to go towards the direction of disorganization. Moreover, the systems can go in both directions in the sense of "integration" and "differentiation".

The underlying principle here is that the augmentation and modification of relationalities in the systems are used to mechanize or implement these directed tendencies.

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

Table 2: Basic Emergent Characteristic of Our System Architecture (DOILS)

Organizing	vs.	Disorganizing
Integrating	vs.	Differentiating

Underlying Mechanism
"Augmentation and Modification of Relationalities in Systems"

of our research is the "total system design of large-scaled complex decentralized artifacts systems attaining **symbiosis** with humans and natural or social environments" [5].

The elapse of time can sometimes be seen in learning, adaptation, development and evolution in living organisms, where differentiation of behaviors (interaction with outside world) plays the essential role (Fig. 3). Differentiated behaviors should then be related to their appropriate contexts of usage, and their integration into a meaningful and purposeful whole should also be attained.

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.

5 Antifacts as Interfacing Media

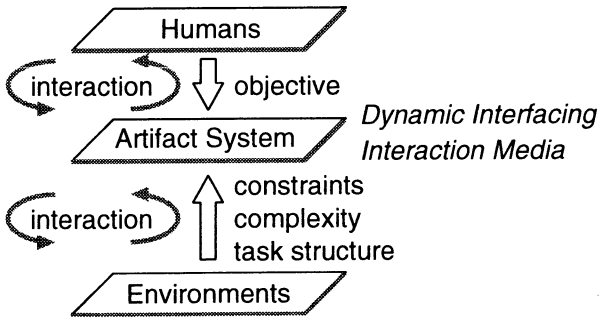


Figure 4: Dynamic Interfacing between Humans and Environments by Decentralized Artifact System

Generally speaking, 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 "interfacing media" which relates objectives (designed goals) to physical world (field) substantiating the objectives. Fig. 4 depicts this interfacing process via communicating the humans' objectives and environmental complexities with each other.

More precisely, as depicted in Fig. 4, the Artifact System can be seen as an Interaction Media for "Dynamic Interfacing" of Humans with Environments by integrating various objectives from the Human side and the constraints, complexities and task structures from the Environmental side.

The meaning of dynamic interfacing is that these entities are put together to be integrated into an organized whole with certain orderness and regularities which will be latterly discussed in relation to the theory of Information Flow [7].

The 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 to 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 field 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" 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 [8] termed such an aspect as a "consensual domain" and a famous philosopher Merleau-Ponty called it "intercorporeite".

6 Conclusions

In this paper, we have discussed the basic standpoint and approaches of our project "System Theory of Bio-Informatics and its Extension towards Engineering Design Theory". Our objective is to design decentralized artifact systems based on the concept of DOILS that can be used as a Dynamic Interfacing Media between Humans and Environments. The structures and the complexities of both sides are integrated into the media thus enabling the dynamic interfacing between both sides. This style of organizing and coordinating interactions among various entities seems to be universal from micro cellular phenomena level to macro man-machine interaction level.

References

[1] M. Dan, *Radings the Complexity of Biological Systems: The Biological Hierarchy*, Heibon-sha, 1996 (in Japanese)

[2] T. Yoshida, *Information Scienceson Self-Organizing Systems*, Shinyo-sha, 1990(in Japanese)

[3] K. Kurokawa, *The Philosophy of Symbiosis*, Academy Edition, London, 1994

[4] K. Minaguchi (ed.), *The Competitive Powens of Japanese Companies*, Tokyo Keizai Shinpo Co. Ltd., 1992 (in Japanese)

[5] O. Katai, Human System Interactions and Intelligent Support - A Tutorial -, *J. of Japanese Society for Artificial Intelligence*, Vol.13, No.3, pp.339-346, 1998 (in Japanese)

[6] J. Flach et al. (Eds.), *Global Perspectives on the Ecology of Human-Machine Systems*, Vol.1, Lawrence Erlbaum Assoc., 1995

[7] J. Barwise and J. Seligman, *Information Flow: The Logic of Distributed Systems*, Cambridge Univ. Press, 1997

[8] T. Winograd and F. Flores, *Understanding Computes and Cognition: A New Foundation for Design*, Ablex Pub. Co., 1986

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Design of Shared Autonomy System Based on Biologically-Inspired Coordination

Tetsuo Sawaragi and **Yukio Horiguchi**
Dept. of Precision Engineering, Graduate School of Engineering, Kyoto University
Yoshida Honmachi, Sakyo, Kyoto 606-8501, Japan.

Email. {sawaragi, yukio}@prec.kyoto-u.ac.jp

Abstract

The most significant and ideal to establish a teleoperation environment for human operators is to let them share the task and enable them to readily activate their own autonomy being conscious neither of the complexity nor the difficulty of the operating task per se caused by intervening artifacts. This is enabled by letting them directly perceive and manipulate objects. There exist some invariant features within the proficient skill-performer's ways of interacting with objects in a VR space. To detect the boundaries of those phases during the actor's performing tasks in real time, we introduce a neural network learning model of Grossberg's Adaptive Resonance Theory (ART). Using this we realize a mechanism that can segment a series of an operator's motion into a number of abstract operation phases and can detect inter-phase boundaries of continuous behaviors. Finally, we present a framework of the reciprocity formed by a human and a robot, and discuss about this in terms of biological coordination.

1 Introduction

Teleoperation techniques are progressively enlarging a human range in which a human operator can feel some direct *empathy* in terms of object size, event duration and control spaces that emerges from human-machine interactions [5], [6]. However, since it is mediated by complex artifacts in between, it is difficult for operators to operate through an intrinsic 'feel' for the process under their control as they do in a direct manipulation task environment. A popular and typical approach to this problem would be an attempt to provide ever greater amounts of information in the display so that it can aid operators' grasping system status [4]. However, system complexity needed for this will soon defeat operators' skilled intuition by the proliferation of physically confusing and ambiguous displays, which causes distancing the operator from the physical cause and effect and moment-to-moment status of the system. This is called a *process empathy* problem. The solution to this problem does exist in how to *ground* the two completely different task environments (i.e., the one where an operator has to interact with the interface and the other one where a teleoperated robot works) with each other by preserving a human's performing intrinsic perception-action skills. Moreover, we consider more a companion and complementary approach in which sharing of effort and action is emphasized, rather than make discrete division of labor between the human and the machine. Related to this, there proposed an idea of "shared autonomy" as a new style of human-machine collaboration [2], which stresses that a human and a machine should collaborate with each other as *equivalent* partners. However, this idea still remains at the conceptual and mainly focuses into the aspects of flexible role-allocation between the human and the machine.

As for such a grounding technique, in this paper we attempt to implement direct display and direct action execution (i.e., embodied interactions) within a virtual reality (VR) by introducing appropriate metaphors and models where needed. Virtual reality is a promising technology in that we can create worlds in which the suspension of laws is arbitrarily tuned and behaviors produced wherein are adjustable in anyway. We are not aiming at using VR to recreate a robot's physical worlds that already exist, but to create a *possible* world (i.e., metaphoric world) that enhances the probability of successful operation of complex system (i.e., a manipulation of a teleoperated robot). VR can be used to modify displacement in space and time, and can return the operator to the proximal region of "human range" from the distal region where his/her teleoperating robot is working. Thus, VR can use all the best facets of human capabilities while allowing the feeling of direct manipulation in human-robot combined system performance spaces. For this purpose, we develop some facilitating modules that can extend the proximate perceptual-action loop established with the VR space to the distal interaction loop joining a human and the real world directly as shown in Fig.1. The distinguished features that our proposing system is aiming at are summarized as follows:

1. Regarding a teleoperation work as a *collaborative work* of a human autonomy (i.e., skills) and a machine autonomy.

2. Regarding an interchange between a human and a robot as an *exchange of events* having some *meanings* (just like as a joint activity of a human-to-human conversation) rather than as a mere transmission of some quantitative signals.

3. Both a human and a robot should share their work environment not only physically but also in a more *task-oriented context* in which they engage.

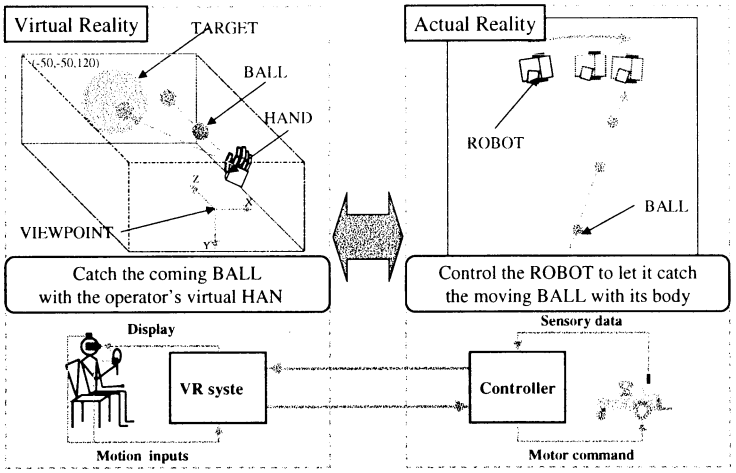


Figure 1: Shared autonomy system using task morphology.

2 Joining Two Perceptual- Acting Loops via a Task Morphology

In living organism systems, reactions to the others and the environment to the individual's exerting action are too complex to predict correctly. Therein, they give up such prediction and change their strategy from predicting others to being assimilated by others. Wherein, sharing some common properties plays a fundamental role [8]. We think it is also the case for the human-robot collaboration; for a natural emergence of collaborative relationships between them, they should be able to recognize and share some common properties. For this we have to identify whether any invariant structures within the human interactions with the task environment exist or not (i.e., invariant features within the proficient skill-performer's ways of interacting with objects in a VR space, in our problem), and if exist, they can be used as a basis for joining two different interaction loops of the human and the robot in a coordinated way. We call such invariant properties as *morphology* of task structures [3].

An idea of morphology or morphism is often discussed by the people of ecological psychology [?]. Vicente who originally proposed an idea of *ecological interface* has also introduced an idea of an *isomorphic mapping* to visualize the invisible for a process control task[9]; a mapping between the geometry of the trapezoid displayed in the interface and the relationships among the variables describing the mass balance as well as the energy balance. The key behind these mappings is that they translate the constraints inherent in the domain into constraints on the interface geometry. This is equivalent to embedding knowledge about domain in the interface. The great advantage of providing an externalized representation of the domain constraints in the interface is that it unburdens the operator's memory and reasoning efforts. Making visible the invisible, then, allows operators to control the system by relying more on the powerful capabilities of perception and action than on comparatively more effortful and error-prone inferential processes. Different from Vicente's task of process control, our teleoperation task requires a bit more complex mapping between the representations of the interfacing VR space and the teleoperated robot; we have to establish a mapping at a *task*-level not only at a *variable*-level. At a variable level, a mapping is done for every moment-to-moment status, while at a task-level a chunk of perception-action series corresponding to a segmented task unit must be mapped preserving the constraints among those segmented tasks (i.e., a morphology of task structures).

To verify existence of invariant features from actual demonstrations of the actors interacting with the object, we at first collect the data of operators' performing a simple task; a task of hitting a moving ball coming bound for him/her by a racket and of making it bounce into a target area within a VR space. A virtual reality space is displayed to an operator through a head mounted display (HMD), in which a moving ball is displayed and a part of an operator's body (i.e., a hand) is also dynamically displayed as he/she moves a hand against the coming ball. Appearance of the ball and the hand is constructed and displayed from an operator's viewpoint. Magnetic 3D position sensors (i.e., a Polhemus 3 space Fastrak System) are mounted on the HDM as well as on an operator's hand, thus their 3D displacements are monitored and are used for the construction of images in a VR space. Varying the velocity of the moving ball Vb from 20 units/sec to 40 units/sec does produce three different task environments, in which tightness of time constraints for an operator is varied. We identified that the interacting profiles of the hand with an object (i.e., a ball) reveal some common or invariant structures consisting of four different phases as shown in Fig.2: Phase A for predicting a ball's trajectory, Phase B when an actor accelerates his hand toward predicted impact point, Phase C when an actor's hand-positioning task is dominant, and Phase D when a hand impacts with a ball. Details of the experimentations are in [7].

3 Segmenting an Operator's Behavior Using ART model

Then, to automate detecting the boundaries of those phases during the actor's performing tasks in real time,

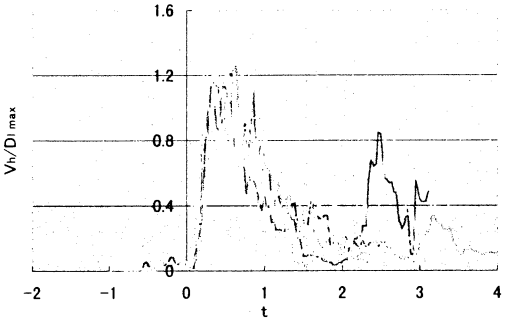


Figure 2: Profiles of translated hand's movements.

we introduce a neural network learning model of Grossberg's *Adaptive Resonance Theory* (ART) [1]. This is a model of unsupervised learning and can organize objects into clusters at a variety of abstraction levels by adjusting its internal parameter called *vigilance* parameter. Using this we realize a mechanism that can segment a series of an operator's motion into a number of abstract operation phases and can detect inter-phase boundaries of continuous behaviors.

As learning of an ART system proceeds, interactions between the inputs and the system generate new steady states, or equilibrium points. The steady states are formed as the system discovered and learns critical feature patterns that represent invariants of the set of all experienced input patterns. The formation of steady states is internally controlled using mechanism that suppress possible sources of system instability. An ART system is sensitive to novelty. It is capable of, without a teacher, of distinguishing between familiar and unfamiliar events, as well as between expected and unexpected events. Within ART, interactions between two functionally complementary subsystems are used to process familiar and unfamiliar events. Familiar events are processed within an attentional subsystem, while the second subsystem called an orienting subsystem plays a role of resetting the attentional subsystem when an unfamiliar event occurs. An ART model functions as an unsupervised dynamic classifier that can generate multigranular categories with a simple adjustment of vigilance parameter. An ART system self-organizes its recognition code, where the environment can also modulate the learning process and thereby carry out a teaching role. Different environments may demand either coarse discriminations or fine discriminations to be made among the same set of objects.

Using the ART model, we attempt to automate the recognition of a human-performing task structure by segmenting a motion sequence into groups corresponding to the four different phases. Herein, a number of task constituents (i.e., four) does not have any significant meaning, but what is more important here would be how to detect inter-phase boundaries from a continuous stream of data that may contain noises. Inputs to ART model are time-series of data collected while a human actor interacts with an object (i.e., a ball) in a VR space by moving a hand against it. These include two variables representing an actor's proprioceptual sensory data, two variables denoting exteroceptual sensory data, and one representing an objective status of an actor's interacting with an object in a VR space. These data are normalized and form a series of a time-temporal vectors from an input data stream. Fig.3 shows a result of ART's classification of an actor's motion sequence, wherein input data streams are overlaid and horizontal bold bars denote classified clusters (i.e., phases of the motion) along the temporal streams. This result is obtained by letting ART model learn with a vigilance parameter of 0.50 that was found in a trial and error way. The result is actually reproducing a transition of the phases that was observed in the human behavior of Fig.3.

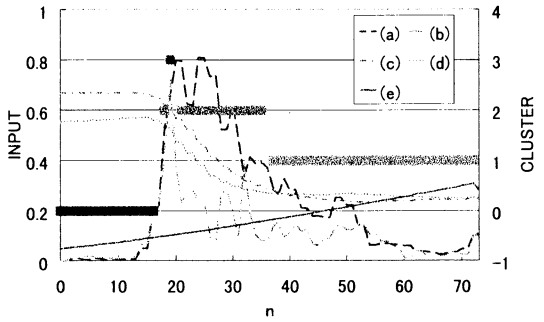


Figure 3: Segmented task by ART model.

4 Collaboration via Boundaries of Segmented Motions

As mentioned in the introductory section, we regard an interchange between a human and a robot as an exchange of "events" having some meanings rather than as a mere transmission of some quantitative signals. As a definition of such an event, we adopt segmented subtasks corresponding to individual phases, and we design a VR space so that it is customized to the individual phase. The main design issue therein is a time-related issue caused by delays in displaying fedback information in response to an operator's executing responses. We analyzed the effects of delays in displaying fedback information and concluded that intervening VR space would do harm to his/her skills and some "aided eyes" are needed.

Then, in order to make a task environment intervened via a VR space more naturalistic, we attempt to vary the ways of displaying information on perception both of proprioceptual and exteroceptual sensory data according to the ongoing task phases. We deform the information displayed within a VR space in a particular way. As mentioned before, during phase C an operator much depends upon the feedback information, which means the effects of delay in displaying sensory consequences do harm to the task performance. Therefore, during this phase the transmission of an operator's hand movement to the actual reality is quitted, and local feedback loop is constructed between an operator and a VR. Detecting a hand starting to move, our system predicts and constructs a posture of a hand in an actual reality, which is presented to an operator in realtime during he/she moves a hand. In this way, our

proposing system does contribute to burying a gap of the two different perceptual-acting cycles having different time spans and evolves asynchronously. We verified that the method contributes to making an operator experience a naturalistic task environment even under a task environment where a large delay is inevitable.

5 Extension towards the Human-Robot Collaborative System

Towards the realization of a shared autonomy system, we proceed to the task analysis of the robot. For this purpose, we at first make an experiment to see how well the proficient operator can control this robot by monitoring continuous visual sensor's images and by commanding movement and steering instructions through a keyboard (i.e., a teleoperation without being intervened by VR). From this result, we can notice that a transition of four different phases does exist revealing a morphological task structure with the result of a human-VR interaction. Although usable perceiving and actuating devices are completely different between those two tasks, a common task structure is shared. This suggests that we would be able to design an ecological interface within which a human- and robot-facilitators interact with each other and form themselves reciprocally under boundary conditions of that common invariant task structure.

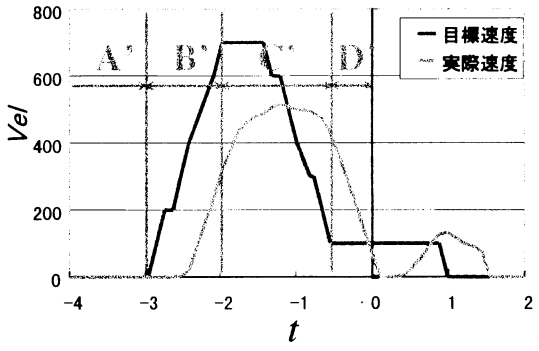


Figure 4: Task structure observed in visually-guided remote control task.

The current status of our system is still too much human-autonomy oriented. Towards the realization of the true shared autonomy system, we have to install the same mechanism also to the machine autonomy side; an ART model monitoring a machine autonomy should be capable of automating detection of its own task structure, and some mutual-adjustment mechanism must be added so that it can automatically establish the isomorphic mapping between the machine and the human autonomy. This interaction would be easily implemented by merely letting the models exchange the vigilance parameters so that both of the perceptual-acting cycles (i.e., boundaries of task phases) are synchronized exactly just at the critical time points. Thus, we will be able to realize a true shared autonomy as a joint activity by the human and the robot.

6 Conclusions

Based on the ecological view regarding a human as a living organism, we can conclude that the fundamental issues of the interface design for the teleoperated robots is to explicitly relate the *interior degrees of freedom* of the human operator to the *exterior degrees of freedom* existing within the environment where the robot operates. Note that this relationship between interior and exterior degrees of freedom will be reversed from the machine autonomy's perspectives, and the reduction of degrees of freedom must be done in a reciprocal fashion and/or based upon a *cyclic causality* rather than a conventional linear causality. Thus, a design of shared autonomy will be modeled as a joint activity model for intention sharing just like as a 'conversation' activity between the humans is so. Anyway, the interface must be designed so that it can absorb the degrees of freedom (or a variability) both from the actual world and from the human operator, and we think that a morphological mapping in between contributes much to this purpose.

References

[1] Grossberg, S.: Competitive Learning: From Interactive Activation to Adaptive Resonance, *Cognitive Science*, **11**, pp.23-63, 1987.

[2] Hirai, S.: Theory of Shared Autonomy, *Journal of the Robotics Research in Japan*, **11**-6, pp.20-25, 1993. (in Japanese)

[3] Holland, J. et al. (Eds.): Induction, The MIT Press, 1986.

[4] Kheddar, A., Fontaine, J.G., Coiffet, P.: Mobile Robot Teleoperation in Virtual Reality, *Proc. of IMACS and IEEE/SMC Multiconference on Computational Engineering in Systems Applications: Symposium on Signal Processing and Cybernetics*, Vol.L, pp.104-109, Tunisia, 1998.

[5] Sawaragi, T., Shiose, T., Akashi, G.: Design of Ecological Interface for Mobile Robot Teleoperation Using Human Naturalistic Reflex between Stimulus and Response, *Proceedings of IEEE International Workshop on Robot and Human Communication (ROMAN'98)*, Vol.1, pp.115 - 120, 1998.

[6] Sawaragi, T., Shiose, T., Akashi, G.: Foundations for Designing Ecological Interface for Mobile Robot Teleoperation, *the Journal of Robotics and Autonomous Systems*, 1999 (in print).

[7] Sawaragi, T., Horiguchi, Y.: Analysis of Task Morphologies for Networking Behavioral Skills via Virtual Reality (to be appeared in special issue of ACM Intelligence Magazine, 2000.)

[8] Ulrich, H. and Probst, G.J.B.: Self-Organization and Management of Social Systems: Insights, Promises, Doubts, and Questions, Springer-Verlag GmbH & Co. KG, 1984.

[9] Vicente, K. and Rasmussen, J.: The Ecology of Human-Machine Systems II: Mediating "Direct Perception" in Complex Work Domain, *Ecological Psychology*, **2**-3, pp.207-249, 1990.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Development of Virtual Collaborator as an Innovative Interface Agent System between Human and Plant Systems: Its Framework, Present Status and Future Direction

H. Yoshikawa, H. Shimoda, W. Wu, H. Ishii, K. Ito
Graduate School of Energy Science
Kyoto University
Uji, Kyoto, Japan, 611-0011
{yosikawa, shimoda, wuwei, hirotake, k-ito}@uji.energy.kyoto-u.ac.jp

Abstract

The objective of this study is to develop "Virtual Collaborator" as a new type of human-machine interface, which works as "intelligent interface agent" to help machine operators controlling large scale machine system such as power plants. In this paper, first, triple dynamics as the design framework of the virtual collaborator is explained, which is the integration of layered module structure and the two scientific frames of brain science and cognitive science. At the present status, a prototype system of the virtual collaborator has been under development, where the prototype system has been successfully constructed based on the triple dynamics. Major functions of the present virtual collaborator are that when an anomaly occurs in a virtual nuclear power plant, the virtual collaborator can autonomously detect it, diagnose the root cause, and operate the control panel in accordance with emergency operation procedures, although it has no bi-directional communication functions. In order to realize bi-directional communication as the next stage development, the relationship between human and the virtual collaborator, and the method of utilizing voice communication are discussed in this paper.

Keyword : virtual reality, nuclear power plant, agent-based interface, verbal communication

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 operation of the machine systems has become more difficult task for users. Especially, in the field of aircraft and power plants, the tendency is conspicuous and an operation error may cause serious accidents. Therefore, the study on the man-machine interface has been extensively made to improve the relationship between human and the machine system.

The main theme of this study is the pursuit of "What is an ideal man-machine interface for large scale system?" The authors pay attention to communication ability of humans with each other which humans originally have, and try to realize an ideal man-machine interface by an agent robot as an avatar of machine system, which behaves and communicates with humans like humans do.

In this study, the agent robot is named "virtual collaborator", which is realized in virtual reality space. It has human-shaped body and can listen, talk, think, behave and collaborate with real humans in operating large and complex machine systems.

Although the authors have not completed developing the whole system of the virtual collaborator, a prototype system was made up as the first stage of the development, in which the virtual collaborator can behave just like a plant operator in the simulated control room of a nuclear power plant (NPP) in virtual reality space.

In this paper, the basic framework of the virtual collaborator, the configuration of the prototype system at the present stage of its development and the future works toward realizing bi-directional communication are described.

2. Basic Framework of Virtual Collaborator

2.1 Layered Module Structure

The design concept of interfacing media as an distributed artificial system interfacing between human and environment is a layered modules structure which consists of three layers, M (Meta-level), S (Structure level), and B (Behavior level) interface E (Environment) as shown in Figure 1[1]. From M to S and B, the integration module as a relation generator for changing relevance of decisions is located. From M to B and E, the module for changing relevance of tasks is located.

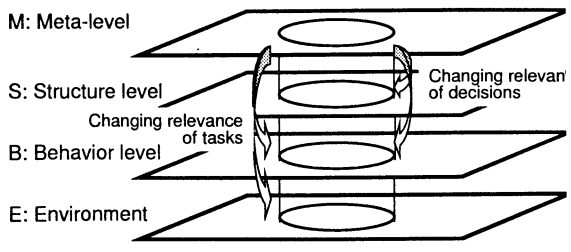


Figure 1: Layered Structure Model.

2.2 Human Model in Views of Brain Science and Cognitive Science

The origin of "human consciousness" has been recently paid attention in the field of brain science, where human consciousness is considered in three levels: arousal (biological consciousness), awareness (consciousness of perception and motion) and self-consciousness (recursive consciousness). The arousal is the basis of human being for behaving with consciousness. The synergistic behavior by the awareness and self-consciousness will form salient characteristics of human being as an advanced and active creature. On the other hand of brain science, both the self-consciousness and awareness can be also interpreted by cognitive paradigm where the consciousness can be interpreted by the synergism of three human memory functions. That is to say, the self-consciousness of attentive, sequential processing corresponds to FWM (Focal Working Memory) as the meta-level, while the awareness of smooth automatic coordination of sensing and motor action is supported by the consolidated works of sub-conscious PWM (Peripheral Working Memory) as the structure level and non-conscious LTM (Long Term Memory) as the behavior level. (See Fig. 2 of integration of brain science and cognitive paradigm.)

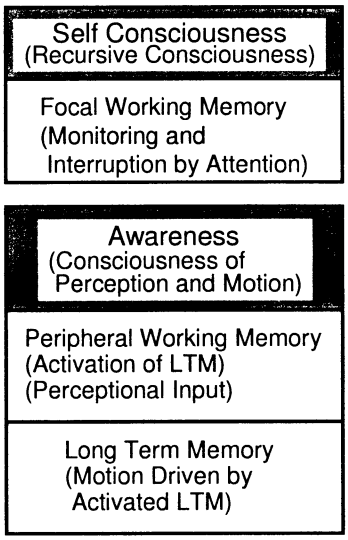


Figure 2: Integration of Brain Science and Cognitive Paradigm.

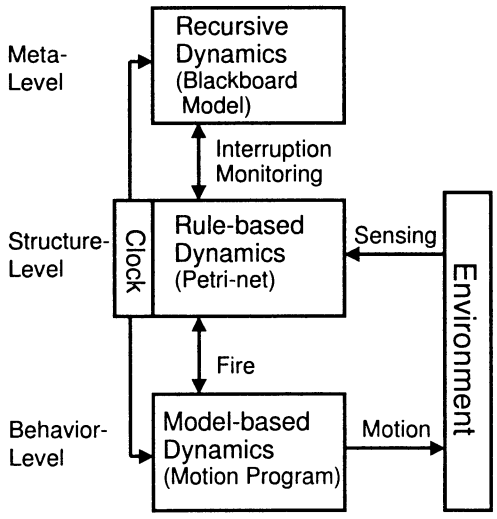


Figure 3: Triple Dynamics System.

2.3 Triple Dynamics

The human model as an intelligent function of the virtual collaborator is modeled by (i) mutually potentiating effect of PWM and FWM, (ii) LTM as an enormous knowledge-based storage which supports unconscious processes, and (iii) selective input channels to the memory of external information via perception system and internal information input to PWM by recalling from LTM[2, 3]. At present, the virtual collaborator has been developed as the driver system for visualizing the operation behavior in virtual reality space.

The driver system of virtual collaborator is shown briefly in Figure 3, with respect to the layered module structure. It is configured as a triple dynamics system that consists of (1) recursive dynamics as the meta-level, (2) rule-based dynamics as the structure level, and (3) model-based dynamics as the behavior level. In short, blackboard model, Petri-net model, and motion program are the models of the recursive dynamics, the rule-based dynamics and the model-based dynamics, respectively. These dynamics modules are driven being synchronized by the clock of the rule-based dynamics with generating mutual relationship and interaction with external environment. The whole configuration of the driver system of virtual collaborator can be depicted as shown in Figure 4. This is one solution how the virtual collaborator as a sort of autonomous robot can escape from "Frame Problem" and "Bernstein Problem" which autonomous robot normally encounter in actual implementation.

3. Prototype System

As the first integration, the prototype virtual collaborator who can behave just like a plant operator in the control room of a nuclear power plant has been developed by applying the concept of the triple dynamics. Although, at present, this prototype system has no bi-directional communication function with human, it can verbalize what it is going to do. The present capability of the prototype virtual collaborator is like the following: when an anomaly occurs in a nuclear power plant, the virtual collaborator can detect it, diagnose the root cause of the anomaly by checking the instrument signals on control panels, and then manage the plant condition to the safety shutdown state. This is like an ideal skilled operator, but the prototype system may behave as if he were a real plant operator who has "fallible" nature. It means that the virtual collaborator does not acts as a perfect operator but he can simulate individual

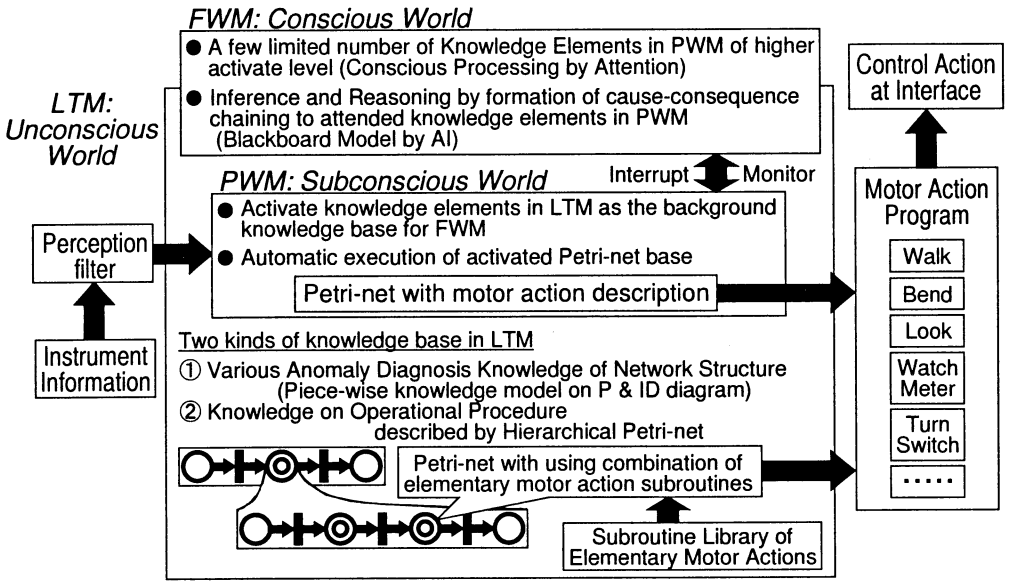


Fig.4: Whole Configuration of Driver System of Virtual Collaborator as Triple Dynamics System.

variations, occasional variations, human uncertainties, human errors, based on realization of the triple dynamics. In addition, his behavior is visualized in 3D virtual space and his thinking process is verbalized by synthesized voice.

Figure 5 shows the experimental configuration of the prototype virtual collaborator. As shown in Figure 5, the whole prototype system is constructed as a distributed real-time system which consists of the following five subsystems: (1) Nuclear Power Plant Simulator, (2) Man-Machine Interface Simulator, (3) Virtual Collaborator Driver System, (4) Human Body Motion Simulator, and (5) Virtual Space Drawing Process. After conducting some simulations by the prototype system and analyzing the simulation results, it was confirmed that the prototype system could simulate behaviors of skilled human operator well in real time, and could visualize the behaviors in the 3D virtual control room.

At the present stage of the prototype system development, it can be utilized as follows by its audio-video information visualizing functions in 3D virtual space.

- The effectiveness evaluation method of the whole man-machine system designing, and
- The economical training system for instructing the trainee how to operate the man-machine interface comprehensively.

For more details, refer to [4].

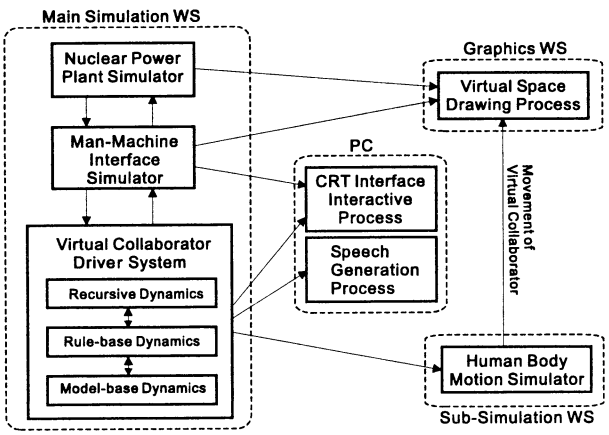


Figure 5: Configuration of Prototype System.

4. Toward Bi-directional Communication of the Virtual Collaborator

In the next stage, the authors aim at extending the function of virtual collaborator as a humanized interface agent. In order to realize such intelligent interface agent, the communication function with human is needed. In this chapter, the authors describe the overview of the bi-directional communication from the viewpoint of relationship between human and interface agent, and discuss the way how to realize bi-directional interaction by voice communication.

4.1 Relationship between human and interface agent

When considering the communication pattern between humans for their cooperative work, the role allocation between them are decided by their relative positions and the contents of the work. Therefore, the role allocation between human and interface agent should be also decided by the same relationship. Concerning the relationship between human and interface agent for nuclear power plant operation, the following three modes can be supposed:

(i) Agent plays the role of a teacher, and instructs the operations to human,
(ii) Human plays the role of a teacher, and in-

structs the operations to agent, and

(iii) Both agent and human cooperate with each other.

As for the mode (i) where machine teach human, various CAI(Computer Assisted Instruction) systems have been studied. With respect to such CAI, the authors have been separately developing an adaptive CAI system, which teaches diagnosis knowledge of nuclear plant anomaly to the student [5,6]. The system can estimate student's thinking process by his operational history and utterance, detects his error, and presents the effective critiques.

As for the mode (ii), the main function is machine learning. It means how machine will be effectively taught to understand and memorizes the contents which human teaches. There are various technological studies such as how to describe knowledge, addition of knowledge and change of knowledge framework, et al. With respect to the virtual collaborator in concern, the virtual collaborator has originally basic knowledge, and the point about (ii) is how the virtual collaborator can increase his knowledge in accordance to its built-in knowledge framework.

In this study, the authors deal with the problem of (iii) that human would cooperate with machine. The target machine system is nuclear power plant.

In this case, the role of the interface agent should be decided based on the share of work and responsibility with human operator. This means that it is necessary to discuss the degrees of automation conducted by the interface agent. In this respect, Sheridan, T. shows Table 1 as the scale of the degrees of automation [7].

From the top of the Table 1, the machine does nothing, offers alternatives, suggests, execute with notification, and acts autonomously at the bottom.

At present, it is difficult to realize full automation in the fields of nuclear power plant and aircraft because it requires high level of safety and reliability to the agent. Therefore, it is important to define each role of human and machine as the degree of automation, and in view of 'Human-centered automation' to 'Keep human in the control loop', the degree of automation should be from 3 to 7 in Table 1.

4.2 How should be bi-directional communication experiment by virtual collaborator

Taking into consideration of the present form of operation in the nuclear power plant, there are two way to share the roles between the virtual collaborator and human as follows;

(1) Human plays a role as a shift-supervisor whereas the virtual collaborator as a operator, and

(2) Human plays a role as a operator whereas the virtual collaborator as a shift-supervisor.

In the case of (1), human as a shift-supervisor commands the virtual collaborator. And the virtual collaborator operates the control panel as a plant operator according to the given order.

In the case of (2), the virtual collaborator as a shift-supervisor offers suggestion to the human operator. And human operates the control panel in accordance to the virtual collaborator's suggestion.

In this study, the authors deal with the case (1) that the human operator acts as a shift-supervisor and the virtual collaborator as an operator, in accordance to the concept of 'Human-centered automation'. In the next section, the authors discuss the way how to inform the human's intention to the virtual collaborator and how to generate the response in order to realize such function of the virtual collaborator.

4.3 Toward realizing bi-directional communication by voice

In this section, the element technology is described in order to realize the function of bi-directional communication between human and machine.

As mentioned above, the virtual collaborator acts as a plant operator who is a cooperative worker with human shift-supervisor. Therefore, the virtual collaborator should be the 'trustful' operator.

The formation of trust is based on durability of performance, technical competence and responsibility, and its attributes are reliability, robustness, familiarity, understandability, explication of intention, usefulness and dependence [8]. In order to conform "trustworthiness" between the both, 'situation awareness' for human

Table 1: Scale of degrees of automation [7]

1.	The computer offers no assistance, human must do it all.
2.	The computer offers a complete set of action alternatives, and
3.	narrows the selection down to a few, or
4.	suggests one, and
5.	executes that suggestion if the human approves, or
6.	allows the human a restricted time to veto before automatic execution, or
7.	executes automatically, then necessarily informs the human, or
8.	informs him after execution only if he asks, or
9.	informs him after execution if it, the computer, decides to.
10.	The computer decides everything and acts autonomously, ignoring the human.

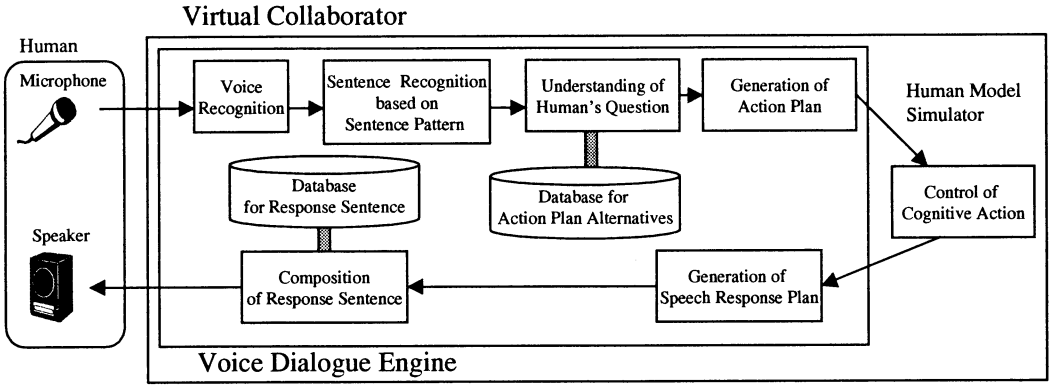


Fig. 6: A Software Organization Proposed for Bi-directional Communication.

is the most important issue, which is the ability to show what the machine is doing and what is going on the system. This ability is related to the attribute 'understandability'.

To sum up, the virtual collaborator should have the function to understand the command of human operator, and the function to present the collaborator's behavior clearly.

As the communication form between human and the virtual collaborator, various methods can be considered. Because the virtual collaborator is a personalized agent interface, the communication method should be familiar to human by simulating the communication between humans. In the case of human communication, there are two forms, verbal and nonverbal communication. The nonverbal communication such as facial expression, gesture and so on plays an important role to transfer the context.

At present, the prototype system of the virtual collaborator has a function to verbalize thinking process[9]. The virtual collaborator utters his intention by synthesized voice in order to present his thinking process explicitly. The utterance corresponds to "think aloud protocol" spoken by subjects in cognitive psychology experiments. The contents of the utterance are the first symptom, hypothesis of root cause, judgement of refereed parameter, and so on, which are necessary to diagnose a plant anomaly occurred in the nuclear power plant.

In order to realize bi-directional communication function, several elemental issues should be studied. They are classified into two questions:

(a) Will the virtual collaborator understand the contents of human utterance? and
(b) Will the virtual collaborator respond and answer back to human timely?

With respect to (a), the utterance of the human operator should be first recognized as a text by voice recognition system, analyzed the text, and then the human's intention, the context, and the peripheral situation are estimated and understood from the analysis result,. Otherwise, the virtual collaborator cannot generate proper response plan.

With respect to (b), the virtual collaborator should first generate his response plan, and then construct the sentence to be uttered by synthesized voice. All of the above processes should be performed in real time.

In order to realize such processes the contents should be limited based on the expectation of human for the virtual collaborator. Then, sentence should be generated from the result of voice recognition based on sentence pattern. The examples of human's request for the virtual collaborator should be the following:

- See the instrument or subsystem in the plant,
- See whether it is normal or anomaly,
- Confirm that it is normal,
- Confirm human's diagnose, and
- Confirm that the root cause by human agrees with the plant status.

After the action generation for the instrument or subsystem, the request should be sent to the human model simulator as a cognitive engine of the virtual collaborator. And, the response of the virtual collaborator should be generated.

Figure 6 shows a tentative software organization plan needed for realizing such bi-directional communication. The authors will conduct the experimental studies to realize bi-directional communication function.

5. Conclusion

In this paper, the basic concept of the virtual collaborator was explained, then the configuration of the prototype system was described based on the triple dynamics. At present, this prototype system has no bi-directional communication function with human, however, when an anomaly occurs in a virtual nuclear power plant, the virtual collaborator can detect it, diagnose the root cause and operate the control panel in accordance with emergency operation procedures. Towards realization of bi-directional communication, the relationship between human and the virtual collaborator was discussed from the viewpoint of the degrees of automation. As the operation assistant of nuclear power plant, the virtual collaborator should play a role of an operator who is

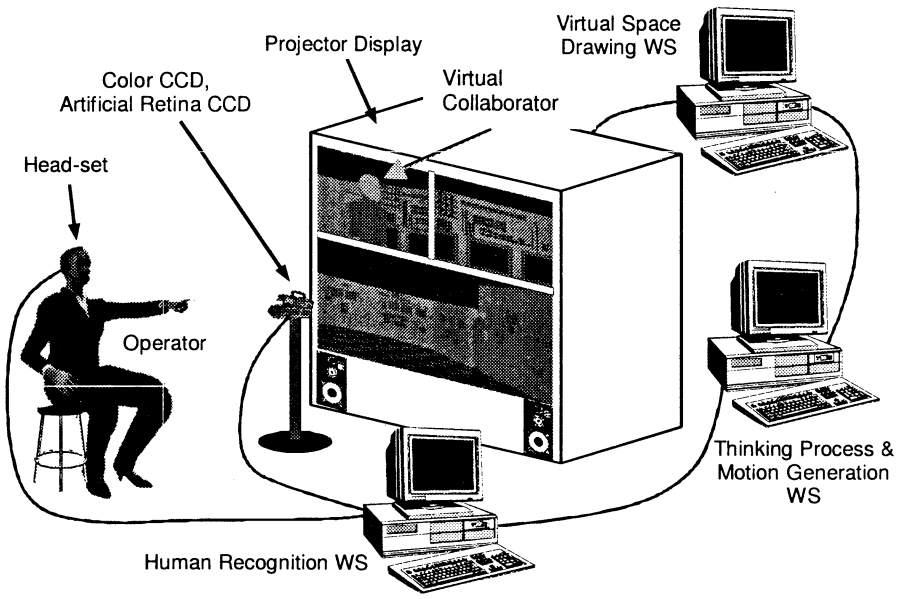


Figure 7: Whole System Set-up for Affective Interface Experiment between Human and Virtual Collaborator.

ordered by human shift-supervisor. At last, the concrete communication method was concerned. As the first step, the authors aim at realizing voice communication and discuss about its subjects to be studied. In the future, the authors are planning to conduct the experimental studies based on the affective interface system and to show its possibility of realizing smooth communication between human and machine system, by the experimental set-up as shown in Fig.7.

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]Katai O(1999), Mathematical Principle of Bio-informatics and Expansion into Methodology of Engineering Design — Interaction and Communication — . Open Symposium'99 "Multi-phasic Approach to Bio-informtics Frontier", Research for the Future Program Integrated Fields, Information on Life Systems, Japan Society for the Promotion of Science, (in Japanese).

[2]Yoshikawa H, *et al.*(1997), Development of an Analysis Support System for Man-Machine System Design Information. Control Engineering Practice 5(3): 417-425.

[3]Wu W, Yoshikawa H(1998), Study on Developing a Computerized Model of Human Cognitive Behaviors in Monitoring and Diagnosing Plant Transients. Proc. of IEEE-SMC'98, 1998, pp.1121-1126.

[4]Ishii H, Wu W, Yoshikawa H, *et al.*(1999), A Basic Study of Virtual Collaborator - The First Prototype System Integration, Proc. of the International Symposium on Artificial Life and Robotics(AROB4th), Beppu, Oita, Japan, Jan 19-22, 1999, pp.682-685.

[5]Ozawa T, Aotake Y, Yoshikawa H, *et al.*(1999), An Experimental Study on Implementing Real-time Estimator of Human Cognitive Process into Eye-Sensing Head Mounted Display for Realizing Adaptive CAI for Teaching Plant Diagnosis Knowledge. Proc. Of Human-Computer Interaction'99, Munich, Germany, Aug 22-27, Vol.2, pp.1196-1200.

[6]Aotake Y, Ozawa T, Yoshikawa H, *et al.*(1999), A New Adaptive CAI System Based on Bio-Informatic Sensing : Study on Real-time Method of Analyzing Ocular Information by Using Eye-Sensing HMD and Method of Adaptive CAI System Configuration. Proc. of IEEE-SMC'99, Tokyo, Japan, Oct 12-15, 1999, Vol.3, pp.733-738.

[7]Sheridan T(1992), Telerobotics, automation, and human supervisory control, MIT Press,Massachusetts, pp.357-358.

[8]*ibid.*, pp.346-348.

[9]Shimoda H, Ishii H, Yoshikawa H, *et al.* (1999), A Basic Study on Virtual Collaborator as an Innovative Human-Machine Interface in Distributed Virtual Environment: The Prototype System and Its Implication for Industrial Application. Proc. of IEEE-SMC'99, Tokyo, Japan, Oct 12-15, 1999, Vol.5, pp.697-702.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A Simulation Model to Study an Adaptive Behavior of Fish School under the Dead-locked Status

H. Nakamine
Dept. of Industrial Arts Education
Kyoto University of Education
Fushimi-ku, Kyoto, 612-8522, Japan
nakamine@kyokyo-u.ac.jp

N. Sannomiya
Dept. of Electronics and Information Science
Kyoto Institute of Technology
Sakyo-ku, Kyoto, 606-5858, Japan
sanmiya@si.dj.kit.ac.jp

Abstract

For an autonomous decentralized system, it is a problem how to avoid the deadlocked status by giving the autonomy to each sub-system. The behavior of fish school is considered as an example of an autonomous decentralized system. In the fish school, each individual behaves autonomously according to the information of some neighboring individuals. In our earlier papers, we constructed the model with a mechanism such as each individual recognizes the deadlocked status, and then changes the quantity of information acquired from other individuals. In this model, we used the parameter M in order to indicate the quantity of information. However, the parameter M was decided by the procedure of trial and error. Therefore, in this paper, the autonomous adjusting mechanism for the M is proposed. By utilizing this model, we simulate the fish behavior affected by the box-shaped trap.

1 Introduction

For an autonomous decentralized system and a multi-agent system, it is a problem how to give the intelligence and the autonomy to the sub-system. In this paper, we consider the behavior of fish school as a typical example of the autonomous decentralized system with the ability to adapt itself to environmental variations. That is, each individual fish behaves autonomously while keeping a specific order based on the information from other individuals. In our earlier papers, we constructed such a model that each individual behaves according to the information of M neighboring individuals. As a simulation result, various facts concerning the behavior of fish school were obtained [1].

Furthermore, we simulated the escaping process in the case where the fish school encounters a box-shaped trap [2, 3, 4]. As a result, it is found that once the school enters the trap, the school does not go out of the trap. Hereafter, we refer this phenomenon as deadlocked status. The lack of the diversity in the school makes the deadlocked status appear when the school consists of individuals with almost same characteristics and the quantity of information acquired from other individuals is large. However, the fish school is not able to keep a specific order, when the quantity of information is too small. On the other hand, in the real field, each individual fish forms the school, because the school behavior is important for the strategy of survival.

Therefore, in this paper, we construct the model with a mechanism such as each individual recognizes the deadlocked status, and then decides the quantity of information (referred as the parameter M) autonomously. In this model, the fish school behaves while choosing between the formation and the collapse of order by the self-judgement. We consider this behavior to be the intelligence of fish school.

2 Fish Behavior Model

In this paper, we construct a mathematical model in order to describe the behavior of fish school swimming in the water tank. Let the position and the velocity of individual i be x_i and v_i, respectively, where $x_i, v_i \in R^2$. Then the motions of N_f fish in a school are described by

$$\left.\begin{aligned} \dot{x}_i &= v_i \\ m\dot{v}_i &= F_{i1} + F_{i2} + F_{i3} \\ i &= 1, 2, \cdots, N_f \end{aligned}\right\} \quad (1)$$

where m is the mean mass of the individuals. F_{i1}, F_{i2} and F_{i3} are the forces which cause the motion of individual i. They are explained as follows [4].

The propulsive force F_{i1} gives a mechanism by which the individual fish swims at a constant velocity

(called the characteristic velocity) in the case where no stimulus affects the individual. An individual fish has a character of behaving under the influence of the motion of other individuals. The schooling force F_{i2} gives a mechanism of keeping a proper distance among neighboring individuals and of uniforming the swimming velocity. F_{i2} includes the parameter M_i which means the number of individuals existing near individual i. In other words, M_i means the quantity of information exchange among individuals. An individual fish receives a certain stimulus from the environment. In the case of our model, the wall of a water tank and obstacles are the environmental elements. The force F_{i3} consists of the repulsive and the attractive force given from the elements.

The model parameters are determined by applying the least squares algorithm to the time series data of a water tank experiment. The water tank is rectangular. It is 150cm long and 100cm wide. The water in the tank is 7cm deep. The fish used in the experiment are bitterling (Rhodeus ocellatus ocellatus), range in length from 3 to 5cm. The time series data obtained in the experiment is sampled at the intervals of $\Delta t = 0.5$sec. The number N of data is chosen as 1200 for each individual. In this paper, simulation is carried out by using a model with the parameter values thus obtained. The number of fish for simulation is twenty. Moreover, it is found out that all the individuals of bitterling have almost same characteristics and the school is called a homogeneous school [4].

3 Autonomous Adjusting Mechanism for the Quantity of Information

In this paper, we add an autonomous adjusting mechanism to the model for fish behavior. In this model, each individual fish recognizes the deadlocked status, and then decides the quantity of information, i.e. M_i included in the schooling force F_{i2}, autonomously. In order to recognize the deadlocked status, the average velocity of the individuals, which exist in the neighborhood of each individual, is used. Then, we have as the adjusting mechanism:

$$M_i = \begin{cases} M_i - 1 & \text{for } |\bar{v}_i| < v_l \text{ and } \dot{\bar{v}}_i < 0 \\ M_i + 1 & \text{for } |\bar{v}_i| > v_h \text{ and } \dot{\bar{v}}_i > 0 \end{cases} \quad (2)$$

$$\bar{v}_i = \frac{1}{M_i} \sum_{j \in N(i)}^{M_i} v_j \quad (3)$$

where v_l and v_h are threshold values to change M_i. In (3), $N(i)$ is the subset of $\{1,2,\cdots,i-1,i+1,\cdots,N_f\}$ whose elements consist of the numbers of the individuals existing near individual i, and M_i is defined as the number of elements in this subset. Equation (2) shows the following process. The phenomena that the absolute value of $\bar{v}_i$ is small and decreases with time show the deadlocked status. In this case, we reduce the value of M_i. After fish school evades the deadlocked status, the value of M_i is increased to recover the behavior with a specific order.

When a fish school consists of individuals with almost same characteristics and a small repulsive force among individuals, it is called a homogeneous and cooperative system. This system has the following characters. When the value of M_i is large, it is difficult that the order formed in the school collapses although the diversity of each individual's behavior is lacking. On the other hand, it is known that the value of $M_i \approx N_f/2$ expands the diversity in the school and is excellent at the adaptability to an environmental change [4].

4 Simulation Results

We investigate the fish behavior affected by appearance of a box-shaped obstacle. Figure 1 shows the configuration in the water tank. There are a box-shaped trap and a leading wall in the tank. At one side of the box, the trap has an entrance from which fish can enter and go out freely. It is thought that the fish school entering in the trap receives a large environmental effect because the area in the trap is 5% or less of the entire water tank.

In order to investigate simulation results, the time variation of the order in fish school is calculated. It is defined that the order of fish school consists of the following state, i.e. the angle between the two velocity vectors of an individual fish and the nearest individual is less than 0.1rad. We define $N_g(\leq N_f)$ as the number of individuals under an order.

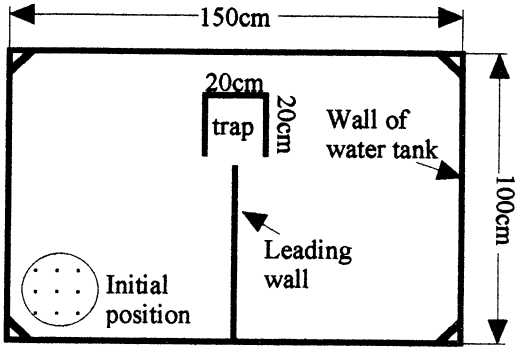
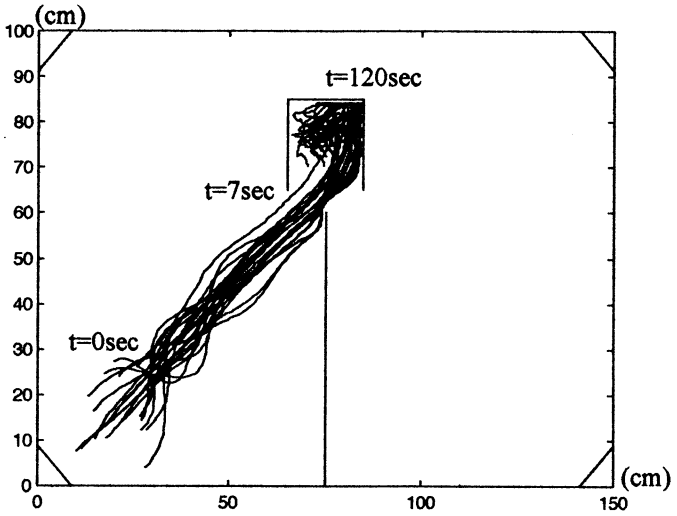


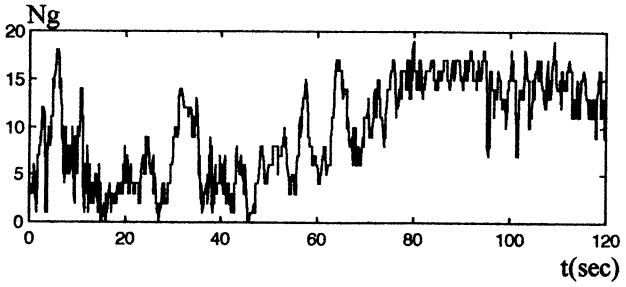
Fig.1 Configuration in the water tank

4.1 The case for fixed value of M_i

In this section, we carry out a simulation in the case where the model has the fixed value of $M_i = 19$. Figure 2(a) shows the swimming trajectory of each individual when the fish school comes across a box-shaped trap. In this case, it is found that all the individuals enter the trap at t=7.0sec. Consequently, The school does not go out of the trap until $t = 120$sec. The order N_g is established locally and increases gradually to $N_g > 15$, and it is difficult to be collapsed as shown in Fig.2(b). In the case of $M_i = 19$, all the individuals concentrate their efforts on the unanimous behavior because of homogeneous school. The high degree of cooperation leads to the deadlocked status of the school.



(a) Trajectories of twenty fish

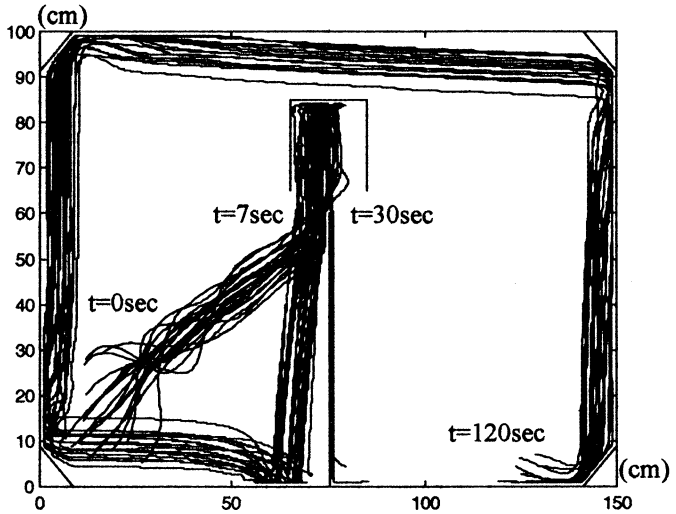


(b) The number of individuals keeping an order

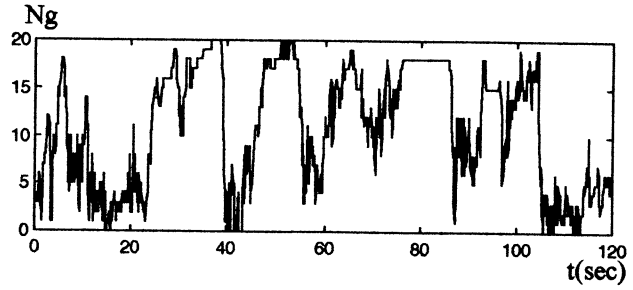
Fig.2 Fish behavior affected by a box-shaped trap (M:fixed)

4.2 The case for varied value of M_i

Figure 3 shows the simulation result for the case where the value of M_i varies with time, where v_l = 0.8cm/sec, v_h = 7.0cm/sec and the initial value of M_i is 19 in (2). It is found from Fig.3(a) that the fish school enters the trap at $t = 7.0$sec. Afterwards a few fish go out of the trap, and then the other fish also go out. Consequently the whole of school goes out of the trap at $t = 30$sec. In addition, the establishment and the collapse of the order are realized easily and are repeated periodically as shown in Fig.3(b). Such phenomena show the diversity of the school is expanded. Thus, it is found out that the model with autonomous adjusting mechanism has adaptability to the deadlocked status.



(a) Trajectories of twenty fish



(b) The number of individuals keeping an order

Fig.3 Fish behavior affected by a box-shaped trap (M:varied)

Table 1 shows the time variation for the distribution of individuals with M_i during $t = 0 \sim 120$sec. As shown in this table, the value of M_i for each individual is decreasing even though the minimum value of M_i is 17. Figure 4 shows the formation of fish school in the box-shaped trap at $t = 25.0$sec, where the symbol "o" is the position of the individual with $M_i = 17$ and the symbol "x" is the position of the individual with $M_i = 18$. It is found from this figure that the value M_i for each individual near the exit of trap is small. Consequently, both a decrease in M_i and an adequate formation are necessary to escape from the deadlocked status.

Table 1 Time variation for the number of individuals with M_i

t(sec)	$M_i = 19$	$M_i = 18$	$M_i = 17$
$t < 20$	20	0	0
$20 \leq t < 22$	0	20	0
$22 \leq t < 36$	0	14	6
$t \geq 36$	20	0	0

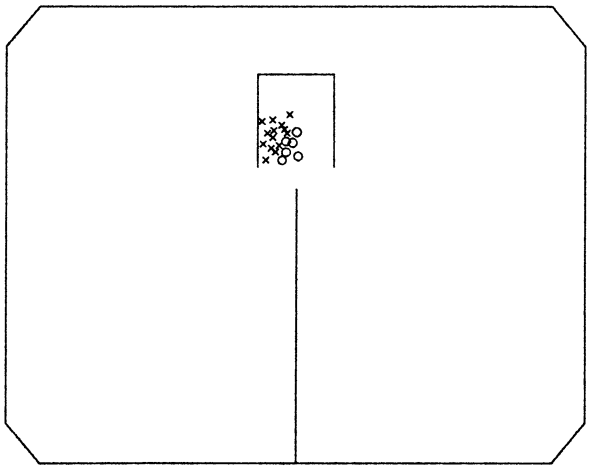


Fig.4 Formation of fish school in the box-shaped trap ($t = 25.0$sec)

5 Summary

In this paper, we proposed the fish behavior model with a mechanism such that each individual recognizes the deadlocked status and then adjusts the value M_i autonomously. By using this model, we investigate the fish behavior affected by a box-shaped trap. As a result, the autonomous adjusting mechanism is effective for the purpose of escape from trap.

In our earlier papers, a modification of the value of M_i varies the diversities of information exchange and decision-making, and then the adaptive behavior of fish school to an environmental variation was simulated. In this paper, the adjusting mechanism for M_i by a top-down approach is transformed into the mechanism by an autonomous approach. For the case where fish school escapes from a box-shaped trap, an adequate value and a distribution of M_i are found out as a simulation result. It is a future problem to clarify the relation between "diversity in the system" and "adaptability as the whole system".

Acknowledgements

This work is partly supported by the Research for the Future Program JSPS-RFTF97I00102 of the Japan Society for the Promotion of Science.

References

[1] Nakamine H and Sannomiya N (1995), "Effect of autonomous decentralized mechanism in fish behavior" (in Japanese). Trans. of ISCIE, Vol.8, No.8, pp.350-356

[2] Sannomiya N and Nakamine H (1998), "A simulation study on cooperative behavior in a fish school". Proc. of the Third Int. Symp. on Artificial Life, and Robotics, Vol.1, pp.17-22

[3] Nakamine H and Sannomiya N (1998), "A simulation study on fish behavior affected by obstacles". Proc. of the 1998 American Control Conference, pp.1090-1091

[4] Tian Y, Sannomiya N and Nakamine H (1999), "A simulation study on adaptability to environmental variations based on ecological systems". Proc. of the 1999 American Control Conference, pp.1759-1763

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Autonomous Decentralized Control of a Quadruped Locomotion Robot using Oscillators

Katsuyoshi Tsujita, Ahmet Onat, Kazuo Tsuchiya and Yousuke Kawano
Dept. of Aeronautics and Astronautics
Kyoto University
Yoshida-honmachi, Sakyo-ku, Kyoto 606-8501, Japan

Abstract

This paper deals with the design of a control system for a quadruped locomotion robot. The proposed control system is based on the bottom up approach and it is composed of a leg controller and a gait pattern controller describing a hierarchical architecture. The leg controller drives the actuators at the joints of the legs using high-gain local feedback control. It receives the commanded signal from the gait pattern controller. The gait pattern controller, on the other hand, involves nonlinear oscillators. These oscillators interact with each other through the signals from the touch sensors located at the tips of the legs. The control system repeatedly moves the legs in predefined motion patterns, synchronizing the motion with the signals from the touch sensor. It also adaptively stabilizes the phase differences between the motions of the legs. The performance of the proposed control system was verified by numerical simulations and hardware experiments.

1 Introduction

Locomotion is one of the basic functions of a mobile robot. Using legs is one of the strategies for accomplishing locomotion. Although simpler forms of locomotion such as wheels can be easier to design and control, using legs for locomotion allows the robot to move on rough terrain, and therefore improves access to many locations. Therefore, a considerable amount of research has been done on motion control of legged locomotion robots. This paper deals with the motion control of a quadruped locomotion robot.

Several gait patterns can be considered for quadruped locomotion robots. The gait pattern in which any combinations of three legs of the robot support the main body at any instant during locomotion is c alled *walk* pattern. For low velocities in which the inertia effect is small enough, the walk pattern is statically stable in terms of the dynamics of the robot mechanism. However, if the velocity of locomotion increases, the locomotion of the robot becomes unstable. On the other hand, *trot* pattern is a gait pattern in which two legs of the robot support the main body at any instant during locomotion. This pattern is statically unstable, and it is difficult for the robot to sustain a stable locomotion at low velocities. However, at higher velocities, the robot can sustain stable locomotion with the trot pattern by using its inertia effectively. Designing a control system for realizing stable locomotion by changing the gait pattern to adapt to the desired velocity or to the properties of the environment is a subject of the research in motion control of the quadruped locomotion robot. In order to design the locomotion controller, we need to design the gait patterns and the motion controller. For designing the gait patterns, we first determine the nominal gait patterns and the trajectories of the legs to realize the nominal gait patterns. The motion controller is designed to realize the nominal motions of the legs.

There are two ways to design the control system of the robot, *top down approach* and *bottom up approach*. The top down approach is based on control theory. The design of the trajectories of the legs and the gait patterns are implemented through optimization based on the inverse model of the robot. The motion controllers are designed based on the linearized model. In this system, nonlinear dynamics such as Coriolis force are eliminated by using the computed torque method, the nonlinear feedback method, etc [1]. The top down approach designs the control system based on the mathematical model of the robot and that of the environment. Therefore, control systems designed in this way are not always robust. On the other hand, the bottom up approach to design the control system is based on the animal behavioral sciences [2] $\sim$ [4]. The animal behavioral science teaches us that the control systems of animals make the legs repeat a forward and backward motion periodically if the legs have no mechanical interaction with the ground. It also teaches us that animals have touch sensors at the tips of the legs and motions of the legs interact with each other through the input signals from the touch sensors. These interactions modify the phase relations of periodic motions of the legs appropriately. As a result, a gait pattern that can satisfy the require-

ments of the locomotion velocity or the properties of the environment emerges. The bottom up approach design is performed in the following way: First, we introduce nonlinear oscillators in the leg controllers and determine the periodic motion of the legs as functions of the phase of the oscillators. Next, we design the local feedback controllers of the legs that use the nominal motions of the legs as reference signals. On the other hand, we determine the dynamic interactions among the nonlinear oscillators so that they interact with each other through the input signals from the touch sensors at the tips of the legs. The phase differences among the nonlinear oscillators emerge through the mutual entrainments of the oscillators. As a result, the proposed control system is expected to generate adequate and stable gait patterns corresponding to the dynamic states of the system or to the physical properties of the environment.

Control systems based on the bottom up approach design have already been applied to the locomotion of hexapod robots in some research. In such research, it has been shown that the control system can adaptively generate the adequate gait patterns corresponding to the system states or to the variations of the environment. The efficiency of the control system was also verified by hardware experiments [5]. On the other hand, it is difficult to design the control system of a quadruped locomotion robot which satisfies the dynamic stability of locomotion. Therefore, it is also difficult to design the control system which can autonomously generate the various gait patterns. There is only a small amount of research based on the bottom up approach that deals with the control system of a quadruped locomotion robot [6].

This paper deals with the design method of the control system of a quadruped locomotion robot based on the bottom up approach. The proposed control system has a hierarchical architecture. It is composed of the leg controller and the gait pattern controller. The leg controller drives the actuators of the legs by using local feedback control. The gait pattern controller involves non linear oscillators. Various gait patterns emerge through the mutual entrainment of these oscillators. The performance of the proposed control system is verified by numerical simulations and hardware experiments.

2 Equations of motion

Consider the quadruped locomotion robot shown in Fig. 1, which has four legs and a main body. Each leg is composed of two links which are connected to each other through a one DOF(degree of freedom) rotational joint. Each leg is connected to the main body through a one DOF rotational joint. The inertial and main body fixed coordinate systems are defined as $[\boldsymbol{a}^{(-1)}] = [\boldsymbol{a}_1^{(-1)},\ \boldsymbol{a}_2^{(-1)},\ \boldsymbol{a}_3^{(-3)}]$ and $[\boldsymbol{a}^{(-1)}] = [\boldsymbol{a}_1^{(-1)},\ \boldsymbol{a}_2^{(-1)},\ \boldsymbol{a}_3^{(-3)}]$, respectively. $\boldsymbol{a}_1^{(-1)}$ and $\boldsymbol{a}_3^{(-1)}$ coincide with the nominal direction of locomotion and vertically upward direction, respectively. Each leg is enumerated from leg 1 to 4, as shown in Fig. 1. The joints of the leg are numbered as joint 1 and 2 from the main body toward the tip of the leg. We define $\theta_i^{(0)}$ $(i = 1,\ 2,\ 3)$ as the components of Euler angle from $[\boldsymbol{a}^{(-1)}]$ to $[\boldsymbol{a}^{(0)}]$. We also define $\theta_j^{(i)}$ as the joint angle of link j of leg i. The rotational axis of joint j of leg i is parallel to the $\boldsymbol{a}_2^{(0)}$ axis. The following equations define the position vector from the origin of $[\boldsymbol{a}^{(-1)}]$ to that of $[\boldsymbol{a}^{(0)}]$.

$$\boldsymbol{r}^{(0)} = [\boldsymbol{a}^{(-1)}]r^{(0)} \quad (1)$$

$$r^{(0)} = [\ r_1^{(0)}\ r_2^{(0)}\ r_3^{(0)}\ ]^T \quad (2)$$

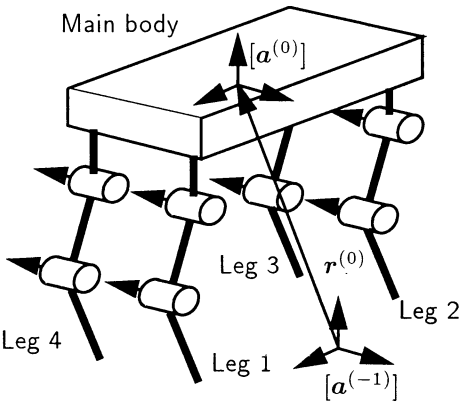


Fig. 1 Schematic model of a quadruped locomotion robot

The state variable is defined as follows;

$$q^T = \left[\ \dot{r}_k^{(0)}\quad \omega_k^{(0)}\quad \dot{\theta}_j^{(i)}\ \right] \quad (3)$$
$$(i = 1, \cdots, 4),\quad (j, k = 1, 2, 3)$$

Equations of motion for state variable q are derived using Lagrangian formulation as follows;

$$M\ddot{q} + H(q,\ \dot{q}) = G + \sum(\tau_j^{(i)}) + \Lambda \quad (4)$$

where M is the generalized mass matrix and the term $M\ddot{q}$ expresses the inertia. $H(q,\ \dot{q})$ is the nonlinear term which includes Coriolis forces and centrifugal forces. G is the gravity term. $\sum(\tau_j^{(i)})$ is the input torque of the actuator at joint j of leg i. Λ is the reaction force from the ground at the point where the tip of the leg makes contact. We assume that there is no slip between the tips of the legs and the ground.

3 Gait pattern control

The architecture of the proposed control system is shown in Fig. 2. The control system is composed of the leg controllers and the gait pattern controller. The leg controllers drive all the joint actuators of the legs as to realize the desired motions that are generated by the gait pattern controller. The gait pattern controller involves non linear oscillators corresponding to each leg. The gait pattern controller receives the commanded signal of the nominal gait pattern as the reference. It also receives the feedback signals from the touch sensors at the tips of the legs. The generated gait pattern is determined by the phase differences between the non linear oscillators. A modified gait pattern is generated from the nominal gait pattern through the mutual entrainment of the oscillators with the feedback signals of the touch sensors. The generated gait pattern is given to the leg controller as the commanded signal of the locomotion pattern of the legs.

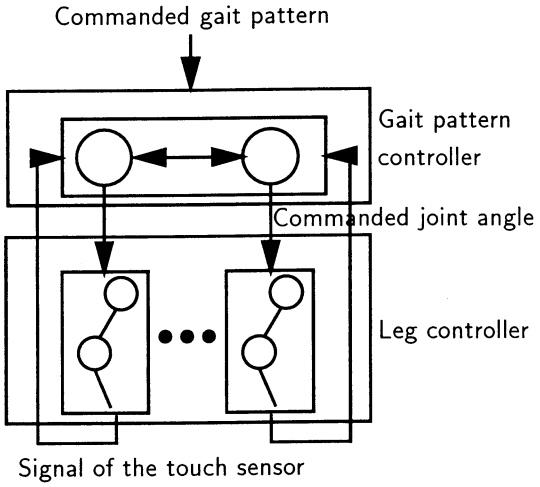


Fig. 2 Architecture of the proposed controller

3.1 Design of the gait

3.1.1 Design of trajectories of the legs

We determine two types of trajectories as the nominal trajectories of the tip of each leg. One is the trajectory for the swinging leg. The other is that for the supporting leg. The position of the tip of the leg where the transition from the swinging state to the supporting state occurs is called the anterior extreme position (AEP). Similarly, the position where the transition from the supporting state to the swinging state occurs is called the posterior extreme position (PEP). We determine the nominal trajectories which are expressed in the coordinate system $[\boldsymbol{a}^{(0)}]$ in the following way: First, we define the nominal PEP $\hat{r}_{eP}^{(i)}$ and the nominal AEP $\hat{r}_{eA}^{(i)}$. The index $\hat{*}$ indicates the nominal value.

The trajectory for the swinging state is a closed curve given as the nominal trajectory $\hat{r}_{eF}^{(i)}$. This curve involves the points $\hat{r}_{eA}^{(i)}$ and $\hat{r}_{eP}^{(i)}$. On the other hand, the trajectory for the supporting state is a linear trajectory given as $\hat{r}_{eS}^{(i)}$. This linear trajectory also involves the points $\hat{r}_{eA}^{(i)}$ and $\hat{r}_{eP}^{(i)}$. These trajectories are given as functions of the phases of the oscillators. The state of the oscillator for leg i is expressed as follows;

$$z^{(i)} = \exp(j\ \phi^{(i)}) \tag{5}$$

where $z^{(i)}$ is a complex number which expresses the state of the oscillator, $\phi^{(i)}$ is the phase of the oscillator and j is the imaginary unit.

The nominal phase dynamics of the oscillator is defined as follows;

$$\dot{\hat{\phi}}^{(i)} = \omega \tag{6}$$

The nominal trajectories $\hat{r}_{eF}^{(i)}$ and $\hat{r}_{eS}^{(i)}$ are given as functions of the phase $\hat{\phi}^{(i)}$ of the oscillator.

$$\hat{r}_{eF}^{(i)} = \hat{r}_{eF}^{(i)}(\hat{\phi}^{(i)}) \tag{7}$$
$$\hat{r}_{eS}^{(i)} = \hat{r}_{eS}^{(i)}(\hat{\phi}^{(i)}) \tag{8}$$

We use one of these two trajectories alternatively at every step of AEP and PEP to generate the desired trajectory of the tip of the leg $\hat{r}_{e}^{(i)}(\hat{\phi}^{(i)})$.

$$\hat{r}_{e}^{(i)}(\hat{\phi}^{(i)}) = \begin{cases} \hat{r}_{eF}^{(i)}(\hat{\phi}^{(i)}) & 0 \le \hat{\phi}^{(i)} < \hat{\phi}_{A}^{(i)} \\ \hat{r}_{eS}^{(i)}(\hat{\phi}^{(i)}) & \hat{\phi}_{A}^{(i)} \le \hat{\phi}^{(i)} < 2\pi \end{cases} \tag{9}$$

The following relationships exist between AEP and PEP.

$$\hat{r}_{eP}^{(i)} = \hat{r}_{eF}^{(i)}(0) = \hat{r}_{eS}^{(i)}(0) \tag{10}$$
$$\hat{r}_{eA}^{(i)} = \hat{r}_{eF}^{(i)}(\hat{\phi}_{A}^{(i)}) = \hat{r}_{eS}^{(i)}(\hat{\phi}_{A}^{(i)}) \tag{11}$$

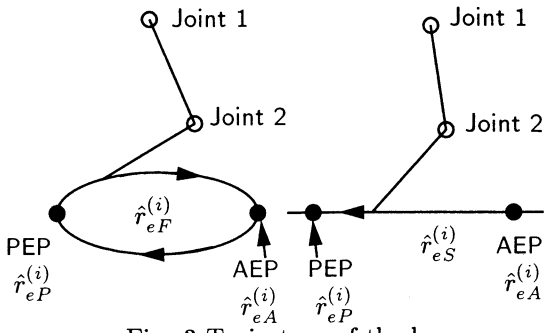


Fig. 3 Trajectory of the leg

Duty ratio $\hat{\beta}^{(i)}$ for leg i is defined to represent the ratio between the nominal time for the supporting state and the period of one cycle of the nominal locomotion.

$$\hat{\beta}^{(i)} = 1 - \frac{\hat{\phi}_A^{(i)}}{2\pi} \tag{12}$$

The nominal stride of leg i is given as follows;

$$\hat{S}^{(i)} = \hat{r}_{eA}^{(i)} - \hat{r}_{eP}^{(i)} \tag{13}$$

3.1.2 Design of the gait pattern

There are three gait patterns in which two legs support the main body at any instant during locomotion: In the trot pattern legs 1 and 3 form one pair and legs 2 and 4 form the other pair, in the pace pattern legs 1 and 2 form one pair and legs 3 and 4 form the other pair, finally in the bounce pattern legs 1 and 4 form one pair and legs 2 and 3 form the other pair.
The following relationships exist among the phase variables of the periodic motions of the legs.

$$\text{Trot} \ : \ \hat{\phi}^{(1)} = \hat{\phi}^{(3)}, \hat{\phi}^{(2)} = \hat{\phi}^{(4)} \tag{14}$$

$$\text{Pace} \ : \ \hat{\phi}^{(1)} = \hat{\phi}^{(2)}, \hat{\phi}^{(3)} = \hat{\phi}^{(4)} \tag{15}$$

$$\text{Bounce} \ : \ \hat{\phi}^{(1)} = \hat{\phi}^{(4)}, \hat{\phi}^{(2)} = \hat{\phi}^{(3)} \tag{16}$$

As can be seen from equations (14) $\sim$ (16), phases of the pairs of oscillators are coupled and the phase differences between them are zero.

A shift of the phase differences between the coupled oscillators by $\frac{\pi}{2}$ causes the gait pattern to change from those explained above to the walk pattern (Fig. 4). For example, the phase shift $\frac{\pi}{2}$ of the trot pattern produces another pattern in which the legs 1,3,2 and 4 touch on the ground in this order. This modified pattern is called *transverse walk*.

In this paper, we consider the gait control method which makes a transition from the trot pattern to walk (or vice versa) by using an external commanded signal.

Trot pattern and walk pattern can be written as follows;

$$\phi^{(i+2)} = \phi^{(i)} + \Gamma^{(m)} \quad (i = 1, 2) \tag{17}$$

$$\Gamma^{(m)} = \begin{cases} 0 & \text{Trot} \quad (m = 2) \\ 2\pi(1 - \hat{\beta}) & \text{Walk} \quad (m = 1) \end{cases}$$

The gait pattern is designed based on Eq.(17).

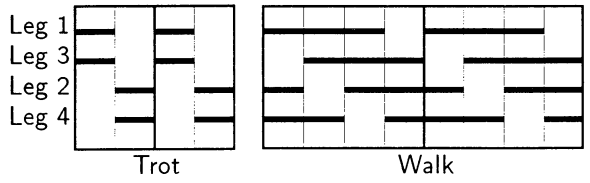


Fig.4. The trot and walk patterns

3.2 Motion control

3.2.1 Leg controller

The angle of joint j of leg i is derived from the geometrical relationship between the trajectory $\hat{r}_e^{(i)}(\hat{\phi}^{(i)})$ and the joint angle. $\hat{\theta}_j^{(i)}$ is written as a function of phase $\hat{\phi}^{(i)}$ as follows;

$$\hat{\theta}_j^{(i)} = \hat{\theta}_j^{(i)}(\hat{\phi}^{(i)}) \tag{18}$$

The commanded torque at each joint of the leg is obtained by using local PD feedback control as follows;

$$\tau_j^{(i)} = K_{Pj}(\hat{\theta}_j^{(i)} - \theta_j^{(i)}) + K_{Dj}(\dot{\hat{\theta}}_j^{(i)} - \dot{\theta}_j^{(i)}) \tag{19}$$
$$(i = 1, \cdots, 4, j = 1, \cdots, 3)$$

where $\tau_j^{(i)}$ is the actuator torque at joint j of leg i, and K_{Pj},K_{Dj} are the feedback gains, the values of which are common to all joints in all legs.

3.2.2 Gait pattern controller

We design the phase dynamics of the oscillators corresponding to each leg as follows;

$$\dot{\phi}^{(i)} = \omega + g_1^{(i)} + g_2^{(2)} \qquad (i = 1, \cdots, 4) \tag{20}$$

where $g_1^{(i)}$ is the term which is derived from the nominal gait pattern and $g_2^{(i)}$ is the term caused by the feedback signal of the touch sensors of the legs.

Function $g_1^{(i)}$ is designed in the following way: We define the following potential function.

$$V(\phi^{(i)}, \Gamma^{(m)}) = K \sum_{i=1}^{2} \left(\phi^{(i)} - \phi^{(i+2)} - \Gamma^{(m)}\right)^2 \tag{21}$$

where parameter $\Gamma^{(m)}$ represents the nominal gait pattern defined in Eq.(17) and is given as a commanded signal.

Function $g_1^{(i)}$ is derived from the potential function V as follows;

$$g_1^{(i)} = -\frac{\partial V(\phi^{(i)}, \Gamma^{(m)})}{\partial \phi^{(i)}}$$
$$= -K\left(\phi^{(i)} - \phi^{(i+2)} - \Gamma^{(m)}\right) \tag{22}$$

Function $g_2^{(i)}$ is designed in the following way: First, we consider the pair of legs i,j ($i = 1$, $j = 3$ or $i = 2$, $j = 4$). Suppose that leg i is in the swinging state and leg j is in the supporting state. The transition of these states occurs at the instant when the swinging leg touches the ground. Suppose that

$\phi_A^{(i)}$, $\phi_P^{(j)}$ are the phases of leg i and leg j at the instant when leg i touches on the ground, respectively. Similarly, $r_{eA}^{(i)}$, $r_{eP}^{(j)}$ are the positions of leg i and leg j at that instance, respectively. When leg i touches the ground, the following procedure is needed.

1. The phase of the oscillator for leg i is changed from $\phi_A^{(i)}$ to $\hat{\phi}_A^{(i)}$. On the other hand, the phase of the oscillator for leg j is changed from $\phi_P^{(i)}$ to 0.
2. Alter the nominal trajectory of the tip of leg i from the swinging trajectory $\hat{r}_{eF}^{(i)}$ to the supporting trajectory $\hat{r}_{eS}^{(i)}$.
3. Replace parameter $\hat{r}_{eA}^{(i)}$, that is one of the parameters of the nominal trajectory $\hat{r}_{eS}^{(i)}$, with $r_{eA}^{(i)}$.
4. Change the nominal trajectory of the tip of leg j from the supporting trajectory $\hat{r}_{eS}^{(j)}$ to the swinging trajectory $\hat{r}_{eF}^{(j)}$. Also replace parameter $\hat{r}_{eP}^{(i)}$, that is one of the parameters of the nominal trajectory $\hat{r}_{eS}^{(i)}$, with $r_{eP}^{(i)}$.

Function $g_2^{(i)}$ is given as follows;

$$\begin{aligned} g_2^{(i)} &= \hat{\phi}_A^{(i)} - \phi_A^{(i)} && \text{at the instant leg } i \text{ touches the ground} \\ g_2^{(j)} &= -\phi_P^{(j)} && \text{at the instant leg } j \text{ takes off from the ground} \end{aligned} \tag{23}$$

The pair of oscillators form a dynamic system that affect each other through two types of interactions. One is continuous interactions derived from the potential function V which depends on the nominal gait pattern. The other is the pulse-like interactions caused by the feedback signals from the touch sensor. Through these interactions, the oscillators generate gait patterns that satisfy the requirements of the environment.

4 Stability analysis of motion

The steady locomotion of the quadruped locomotion robot is strictly periodic and is characterized by a limit cycle in the state space. In this section, the analysis of the stability of the limit cycle in the steady locomotion of the quadruped locomotion robot is described.

The stability of the limit cycle is examined in the following way: First, four variables are selected as state variables.

$$X \in R^4, \quad X = \left[\begin{array}{cccc} \theta_1^{(0)} & \theta_2^{(0)} & \dot{\theta}_1^{(0)} & \dot{\theta}_2^{(0)} \end{array} \right] \tag{24}$$

When the robot starts the locomotion under a certain initial condition, the variable set X moves on a certain trajectory in the four-dimensional state space. If we choose a Poincaré section using the timing when the tip of a leg touches the ground, the first intersection of the trajectory of X with the Poincare section is mapped as X_0, and for every intersection, the corresponding values of X lead to a sequence of iterates in the state space.

$$X_1 \quad X_2 \quad \cdots \quad X_n \quad \cdots$$

The Poincaré map from X_n to X_{n+1} is expressed as follows;

$$X_{n+1} = F(X_n) \tag{25}$$

The fixed point $\hat{X}$ is defined if $\hat{X}$ satisfies the following equation on the Poincaré section.

$$\hat{X} = F(\hat{X}) \tag{26}$$

This Poincaré map is approximated by use of linearization around the fixed point.

$$X_{n+1} - \hat{X} = M(X_n - \hat{X}) \tag{27}$$

The necessary condition for the asymptotic stability of the sequence of points $\{X_n\}$ is that all of the eigen values of matrix M are smaller than one in magnitude, λ_k $(k = 1, \cdots, 4)$ $|\lambda_k| < 1$.

5 Numerical simulation

Table 1 shows the physical parameters of the robot which are used in numerical simulations.

Table 1

Main body		
Width	0.182	[m]
Length	0.338	[m]
Height	0.05	[m]
Momentum of inertia (x)	0.010	[kgm^2]
Momentum of inertia (y)	0.046	[kgm^2]
Momentum of inertia (z)	0.046	[kgm^2]
Legs		
Length of link 1	0.026	[m]
Length of link 2	0.188	[m]
Length of link 3	0.193	[m]
Mass of link 1	0.320	[kg]
Mass of link 2	0.918	[kg]
Mass of link 3	0.595	[kg]
Momentum of inertia of link 1	1.0E-06	[kgm^2]
Momentum of inertia of link 2	1.8E-02	[kgm^2]
Momentum of inertia of link 3	5.1E-03	[kgm^2]

Numerical simulations are implemented under the condition that the nominal stride $\hat{S}$ and the nominal gait pattern are fixed. The nominal duty ratio $\hat{\beta}$ is

selected as a parameter. The dynamic stability of each gait pattern depends on the value $\hat{\beta}$. For example, for $\hat{\beta} = 0.5$, trot, pace and bounce patterns are the most stable dynamically, whereas for $\hat{\beta} = 0.75$ the walk pattern is stable.

The trajectory of the tip of the leg is given as follows;

$$r_{eS1}^{(i)} = \frac{S}{\phi_A^{(i)}}(\phi^{(i)} - \cos\phi^{(i)}) + r_{eP1}^{(i)} \quad (28)$$

$$r_{eS2}^{(i)} = \frac{S}{\phi_A^{(i)}}(1 - \sin\phi^{(i)}) + r_{eP2}^{(i)} \quad (29)$$

$$r_{eF1}^{(i)} = \frac{S}{2\pi - \phi_A^{(i)}}(\phi^{(i)} - \phi_A^{(i)}) + r_{eA1}^{(i)} \quad (30)$$

$$r_{eF2}^{(i)} = \frac{S}{2\pi - \phi_A^{(i)}}(\phi^{(i)} - \phi_A^{(i)}) + r_{eA2}^{(i)} \quad (31)$$

The nominal time period of the swinging state is chosen as 0.20 [sec]. The frequency band width of joint 1, 2 and 3 are given as 50.0 [Hz], 5.5 [Hz], 9.5 [Hz] for feedback gains of the joints, respectively. Numerical simulations are implemented for the walk pattern and trot pattern.

In order to compare the performance of the proposed control system, another control system is simulated in which the leg controller drives the legs as to follow the nominal trajectories using local feedback control. Two cases are examined by numerical simulations: One is the case where the controller uses a fixed gait pattern and the nominal trajectories as the desired trajectories (CASE 1). The other is the case where the controller also uses a fixed gait pattern but the controller is the proposed one that is explained in Section 3 (CASE 2).

Figure 5 shows the result of the stability analysis where we checked the motion of the robot. It is performed by using the method introduced in Section 4, based on the simulation results. It can be seen that the proposed control system has better performance compared with the control system that uses the nominal trajectories as the desired trajectories. The proposed control system established stable locomotion of the robot with a wide parameter variance for duty ratio $\hat{\beta}$.

Figure 6 shows the gait pattern obtained as the simulation result of CASE 1. From these figures, it can be seen that if $\hat{\beta}$ is varied excessively, instability occurs in the fixed gait pattern. For the case where the walk pattern is being commanded, if $\hat{\beta}$ is decreased to 0.5 an unstable and irrhythmic pattern is promoted (Fig. 6.a). Similarly, for the case where the trot pattern is being commanded, if $\hat{\beta}$ is increased to 0.75 the gait pattern is still that of trot (Fig. 6.d). The reason of the instability is that the control system of CASE 1 cannot modify the gait patterns adaptively to the variations in the locomotion velocities.

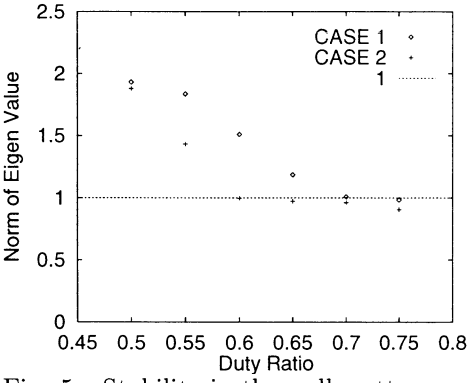


Fig. 5.a Stability in the walk pattern

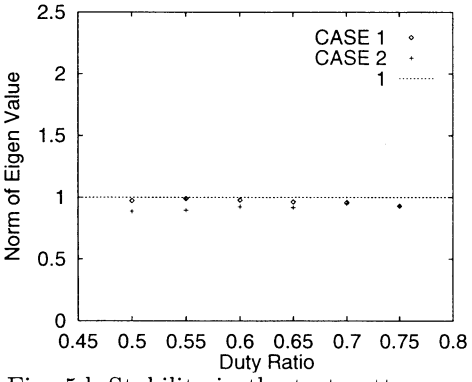


Fig. 5.b Stability in the trot pattern

CASE 1:

Walk

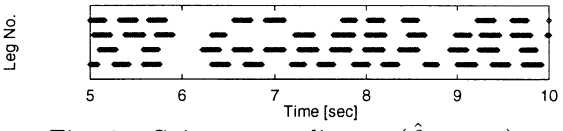


Fig. 6.a Gait pattern diagram($\hat{\beta} = 0.50$)

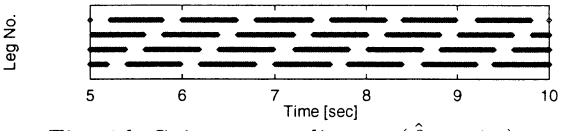


Fig. 6.b Gait pattern diagram($\hat{\beta} = 0.75$)

Trot

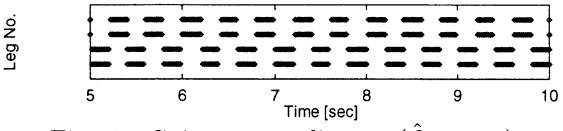


Fig. 6.c Gait pattern diagram($\hat{\beta} = 0.50$)

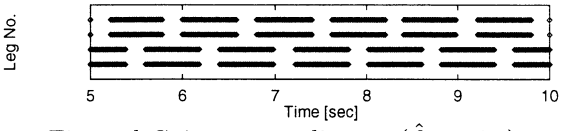


Fig. 6.d Gait pattern diagram($\hat{\beta} = 0.75$)

Figure 7 shows the gait pattern obtained from the simulation result of CASE 2. By varying the duty ratio, the appropriate gait pattern emerges. This is especially evident in Fig. 7.d where the trot pattern is being commanded but $\hat{\beta} = 0.75$. The most stable gait pattern for this duty ratio, the walk pattern emerges. Figure 8 indicates the correlation coefficients among the motion of the legs which are calculated from the results of the simulations where the walk pattern is commanded(Fig. 8.a) and the trot pattern is commanded(Fig. 8.b). The correlation coefficients are given in Eq.(32), which characterize the relationships among the motion of the legs.

From these figures, two results are derived. The first one is that both Fig. 8.a and Fig. 8.b show a similar trend. This means that the controller behaves in a similar fashion regardless of the commanded pattern. The second result is that the correlation coefficient degrades as duty ratio $\hat{\beta}$ increases. However, it is known that an increase in $\hat{\beta}$ implies a decrease in the velocity of locomotion and vice versa. This means, therefore, that the control system modifies the commanded nominal gait pattern adaptively, corresponding to the locomotion velocity and it also preserves the dynamic stability of locomotion. As a result, if the walk pattern is commanded but a high velocity is desired, the control system changes the gait pattern from walk to trot autonomously. The reverse is also true; if the trot pattern is commanded but a slow velocity of locomotion is desired, the walk pattern is generated.

$$\zeta_i = \begin{cases} 0 & \text{Supporting state} \\ 1 & \text{Swinging state} \end{cases}$$

$$W_{ii+2} = \frac{\int \zeta_i \zeta_{i+2} dt}{\sqrt{\int \zeta_i dt} \sqrt{\int \zeta_{i+2} dt}} \quad (i = 1, 2) \quad (32)$$

CASE 2:
Walk

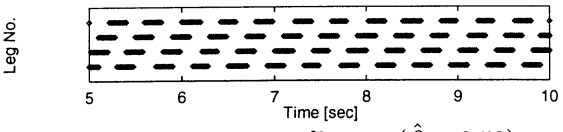


Fig. 7.a Gait pattern diagram($\hat{\beta} = 0.50$)

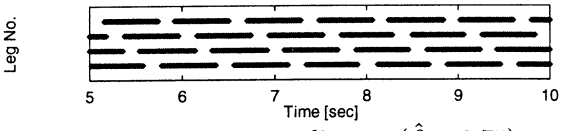


Fig. 7.b Gait pattern diagram($\hat{\beta} = 0.75$)

Trot

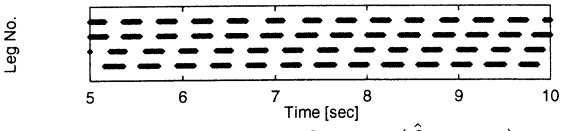


Fig. 7.c Gait pattern diagram($\hat{\beta} = 0.50$)

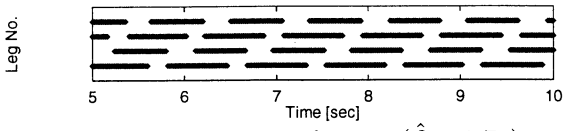


Fig. 7.d Gait pattern diagram($\hat{\beta} = 0.75$)

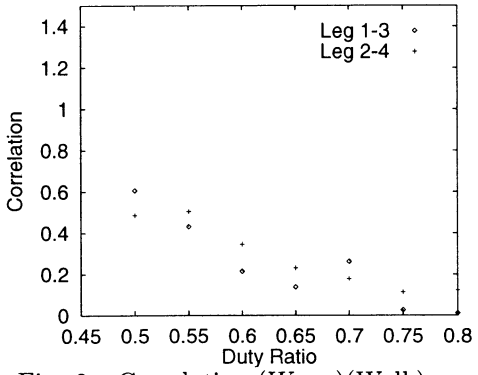


Fig. 8.a Correlation (W_{ii+2})(Walk)

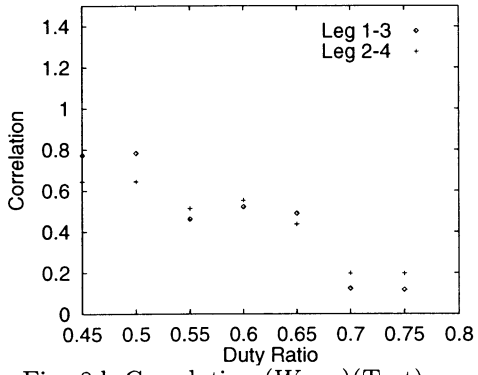


Fig. 8.b Correlation (W_{ii+2})(Trot)

6 Hardware experiments

A picture of the hardware equipment is shown in Fig. 9.

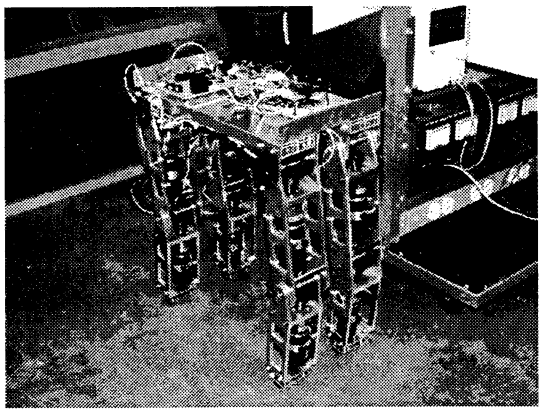


Fig. 9 The hardware equipment

The architecture of the hardware equipment is shown in Fig. 10.

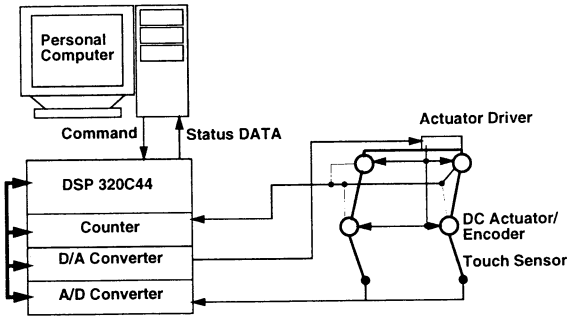


Fig. 10 The architecture of the hardware equipment

The performance of the proposed control system was verified by hardware experiments. Figures 11.a and 11.b show the effect of duty ratio $\hat{\beta}$ when the walk pattern is commanded, and Figs. 12.a and 12.b show the same, in hardware experiments. From these figures, we can see that the proposed control system has a good robustness and adaptability.

Walk

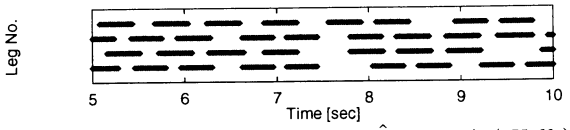


Fig. 11.a Gait pattern diagram($\hat{\beta} = 0.65$) (Walk)

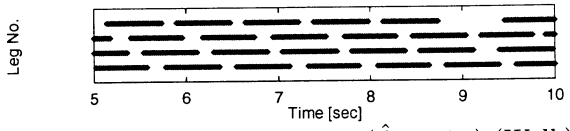


Fig. 11.b Gait pattern diagram($\hat{\beta} = 0.75$) (Walk)

Trot

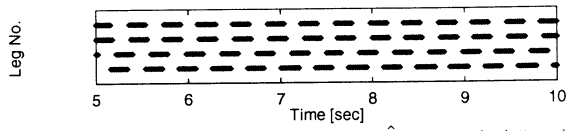


Fig. 12.a Gait pattern diagram($\hat{\beta} = 0.50$) (Trot)

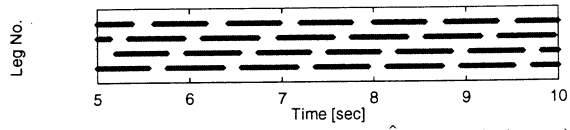


Fig. 12.b Gait pattern diagram($\hat{\beta} = 0.75$) (Trot)

7 Conclusions

We proposed a control system with a hierarchical architecture which is composed of the leg controller and the gait pattern controller. The leg controller drives the actuators at the joints of the legs by use of high-gain local feedback based on the commanded signal from the gait pattern controller. Whereas the gait pattern controller alternates the motion primitives synchronizing with the signals from the touch sensors at the tips of the legs, and stabilizes the phase differences among the motions of the legs adaptively. The performance of the proposed control system was verified by numerical simulations and hardware experiments.

Acknowledgments

The authors were funded by grants from the The Japan Society for the Promotion of Science (JSPS) as the Research for the Future program (RFTF) and from the Japan Science and Technology Corporation (JST) as the Core Research for Evolutional Science and Technology program (CREST).

References

[1] T. Mita and T. Ikeda, "Proposal of a Variable Constraint Control for SMS with Application to a Running and Quadruped," Proc. of the 1999 IEEE International Conference on System, Man and Cybernetics, III, pp.140-145, (1999)

[2] H. Cruse, C. Bartling, J. Dean, T. Kindermann, J. Schmitz, M. Schumm and H.Wagner, "Coordination in a Six-legged Walking system: Simple solutions to complex problems by exploitation of physical properties," From Animals to Animats, pp.84-93, (1996)

[3] G.M.Nelson,R.D.Quinn,R.J.Bachmann and W.C.Flannigan, "Design and Simulation of a Cockroach-like Hexapod Robot," Proc. 1997 IEEE International Conference on Robotics and Automation, No.2, pp.1106-1111, (1997)

[4] G.M.Nelson and R.D.Quinn, "Posture Control of a Cockroach-like Robot," Proc. of the 1998 IEEE International Conference on Robotics and Automation, Vo.1, pp.157-162, (1998)

[5] K. Akimoto, S. Watanabe and M. Yano, "A insect robot controlled by the emergence of gait patterns," Proc. of International Symposium on Artificial Life and Robotics, Vol.3, No.2, pp.102-105, (1999)

[6] H.Kimura, K.Sakaura and S.Akiyama, "Dynamic Walking and Running of the Quadruped Using Neural Oscillator," Proc. IROS, Vol.1, pp.50-57, (1998)

[7] M.J.Coleman, M.Garcia, A.L.Ruina and J.S.Camp and A.Chatterjee,"Stability and Chaos in Passsive-Dynamic Locomotion," Solid Mechanics and its Applications, Vol.63, pp.407-416, (1997)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

How Human Learns to Control a Non-linear System in Iterative Tracking Tasks?

*Muhammad Arif, **Hikaru Inooka and **Tadashi Ishihara
*CCSG, PIEAS, Pakistan,
**GSIS, Tohoku University, Japan

Abstract

In order to apply human control techniques to the design and improvement in the automatic control methods, it is essential to have a good understanding of how a human can be expected to interact with a non-linear system in iterative tasks. In this paper, human capabilities in performing repetitive trajectory tracking tasks are addressed. Time domain characteristics of human learning are studied and some key features of human learning are extracted from these experiments.

1 Introduction

Mastering of a difficult task under diverse environmental conditions is a key feature of human control. Human beings are capable of performing the new tasks successfully after enough practice without having any prior knowledge. For instance, task of a crane operator is a good example of manual control. A novice crane operator feels difficulty in maneuvering the crane but after enough practice, he can control the crane to follow the desired trajectory very smoothly.

Several attempts to investigate some aspects of the man-machine system (MMS) have been reported. An important class of MMS models may be referred to as the control system theoretic models. Historically much research has been done in the manual control, especially in the context of aircraft control. Many researchers have studied the tracking performance of the human operator [1,2] using different techniques. Moreover, the ability of the human operator in performing control tasks and the control characteristics while working as an adaptive controller [3], an optimal controller [4] and telemanipulation [5] have been studied. R. Ikeura and H. Inooka [6] have presented an iterative modification method using the master slave manipulator.

In this paper, we will present some results of the human operator's learning to track a desired reference trajectory and trying to minimize the error between the plant output and the reference input trajectory after each iteration. The plant dynamics are not known to the human operator.

A human operator performs a new trajectory tracking task starting with some initial guess and then in each next iteration, he recalls the previous control pattern and modifies it simultaneously according to the task requirements. Learning of the human operator can be described by the following equation.

$$u_i(t) = \gamma_i u_{i-1}(t) + \delta u_i(t) \qquad (1)$$

The control action of the human operator in the i^{th} iteration comprises of two terms, recalling the previous control action $u_{i-1}(t)$ and a modification term $\delta u_i(t)$ to improve his previous performance in some sense, for instance, decreasing the error between the reference trajectory and the system's output. In this equation, γ_i corresponds to the recalling factor. This recalling factor is introduced to incorporate the fact that the human operator finds difficulty to recall his previous control pattern precisely. But as he performs the task many times, he improves his recalling factor. For an experienced human operator, this factor approaches to one. The modification term $\delta u_i(t)$ corresponds to the modification in the previous control input to improve the tracking or the performance.

M. Arif and H. Inooka [7] have performed some experiments regarding the human performance in iterative learning control and found out that the human operator can perform the task quite intelligently. The important feature they have quoted is that the human operator can not improve his performance if the error spectrum shifts to the higher frequency range.

Apart from the limitation of the human performance due to the shifting of the frequency components of the error and error rate to higher frequencies, the inherent limitations of the human operator also plays main role in his performance. Better visual information perception and a better control sensitivity

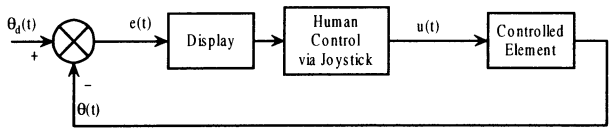


Figure 1: Compensatory manual control setup.

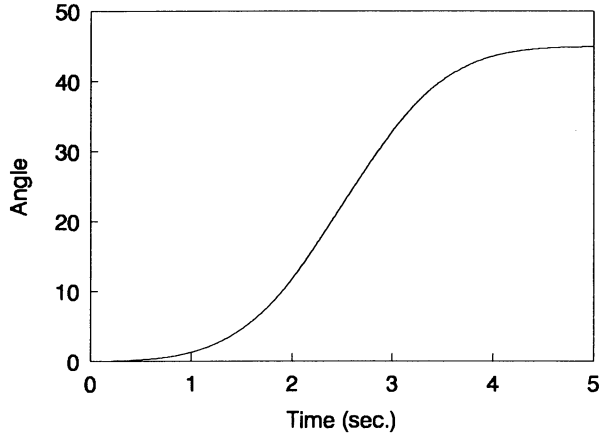


Figure 2: The desired trajectory.

improve the human learning and help him to cope with his inherent limitations[8].

2 Experimental Setup

2.1 Compensatory Manual Control Setup

The compensatory manual control setup as shown in the Figure 1 is equivalent to a one dimensional servomechanism in which the human operator has only one input, the error, the difference between actual system response and the desired or reference trajectory. The majority of the research in manual control theory has dealt with single loop compensatory systems, where the human operator's problem is to observe and compensate the error between the system output and the desired trajectory.

To study the human performance in iterative tracking task, a nonlinear single link pendulum is used as the system. The dynamic equation of the nonlinear model of a single link pendulum is

$$I\ddot{\theta} + \zeta\dot{\theta} + (\frac{1}{2}m + M)gl\sin\theta = \tau \qquad (2)$$

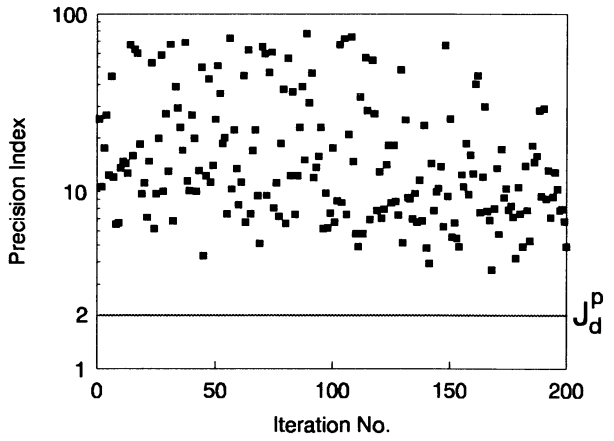


Figure 3: Precision index J^{pi} using compensatory manual control setup.

Where $\theta(t)$ is the angular position of pendulum and $\tau(t)$ is the deriving torque. The variables m, M and l are the mass, tip load and length of the pendulum respectively. I is the moment of inertia with respect to the joint and defined as

$$I = Ml^2 + \frac{1}{3}ml^2 \qquad (3)$$

Let $m = 0.5\ Kg$, $M = 1.0\ Kg$, $l = 1.0\ m$, $g = 9.8\ m/s^2$ and damping term $\zeta = 0.6$ be chosen as system parameters with the sampling period of 0.01 second. $\theta_d(t)$ is the desired trajectory. Precision index J^{pi} of i^{th} iteration is defined as

$$J^{pi} = \max_{0<t\leq T} |e_i(t)| = \max_{0<j\leq N} |e_i(j)| \qquad (4)$$

An analog joystick is used as a human operator control device. Data from the joystick is transferred to the computer through an ADC board (12-bit resolution) at the rate of 100 samples/second.

To illustrate the difficulty of the human operator in performing high precision task, an experiment is performed for the tracking of a desired trajectory as shown in Figure 2[8]. The desired precision index J_d^p is set to be 2^0. The desired final position is set to be 45^0. The compensatory manual control setup as shown in Figure 1 is used for the tracking experiment. The precision index J^{pi} is plotted at log scale in the Figure 3. The base-line set at 2^0 in the figure shows the desired precision index J_d^p. It can be observed from the figure that the human operator was unable to

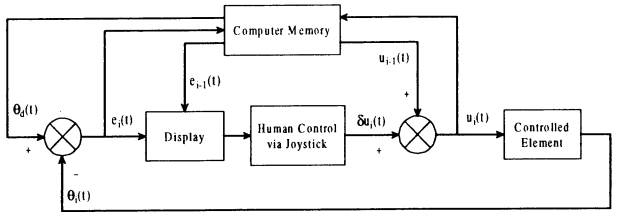


Figure 4: Proposed experimental setup.

achieve the required precision in 200 trials. Based on this observation it is concluded that using such kind of experimental setup it is difficult to perform high precision tracking task by the human operator.

2.2 Proposed Experimental Setup for Iterative Learning

In our setup, we have designed a computer assisted experimental setup based on the iterative learning to study the human learning and performance in the high precision tracking tasks.

Figure 4 shows the block diagram of the experimental setup of computer assisted iterative learning control of a given plant by the human operator. A computer simulates the plant dynamics and the results in term of the error signal are displayed to the human operator. $\theta_d(t)$ is the desired trajectory which is required to be followed by the plant for a fixed time period of $[0, T]$. Error $e_i(t)$ at time t and $e_{i-1}(t)$ for $t \in [0, T]$ are displayed to the human operator on CRT screen and the human operator tries to correct the error using joystick actions. In the i^{th} iteration, observing the resulting error $e_i(t)$, the human operator generates the appropriate signal $\delta u_i(t)$ which is then added to the $u_{i-1}(t)$ stored previously in the memory to generate the control signal input $u_i(t)$. This control input $u_i(t)$ is applied to the controlled element or plant. The error $e_i(t)$ and the control input $u_i(t)$ are also stored in the computer memory to be used in the next i.e. $i + 1^{th}$ iteration. At the end of i^{th} iteration, the precision index J^{pi} is calculated using the data of the error of i^{th} iteration. If the precision index J^{pi} becomes equal or less than the desired precision index J_d^p then the iteration process is terminated. Otherwise the same above procedure is repeated for the next i.e. $i+1^{th}$ iteration. $u_i(t)$ is used as the control input to the system for the i^{th} iteration and also stored in the memory as a modified control signal. Initial conditions remain the same for each iteration.

The effectiveness of the above proposed experimental setup is illustrated by repeating the same experiment as done in the section 2.1 using the compensatory manual control setup. The precision index J^{pi} is plotted in Figure 5. It can be observed from the figure that the human operator has succeeded in performing the tracking task by the desired precision index of 2^0 in 26 iterations. Hence our proposed experimental setup gives us an opportunity to study the human learning and performance in high precision tracking tasks very efficiently.

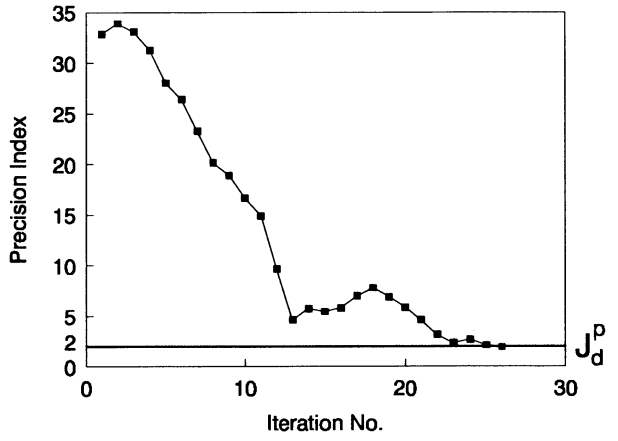


Figure 5: Precision index J^{pi} using our proposed experimental setup.

3 Experiments for the Recalling Factor γ_i

3.1 Experimental Setup

The iterative learning as described by the equation (1) is a function of the recalling factor γ_i and the modification term $\delta u_i(t)$. To get an idea about the recalling factor γ_i associated with the learning, we have done some preliminary experiments using the well known experimental setup for compensatory manual tracking tasks as shown in the Figure 3. The error is displayed to the human operator as a dot moving with time on a CRT screen. The desired trajectory $\theta_{d1}(t)$ is

$$\theta_{d1}(t) = \frac{\pi}{2} \frac{1}{\sqrt{2\pi}a} e^{-b(\frac{t-2.5}{a})^2} \qquad t \in [0, T] \qquad (5)$$

where $a = 0.8$ and $b = 0.5$ and $T = 5$ seconds.

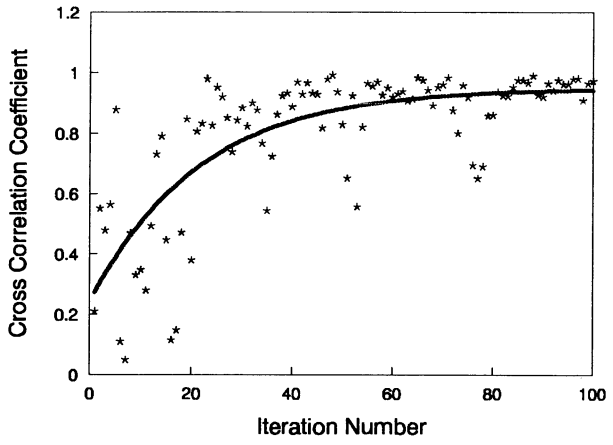


Figure 6: Cross correlation coefficient of $u_i(t)$ and $u_{i-1}(t)$[9].

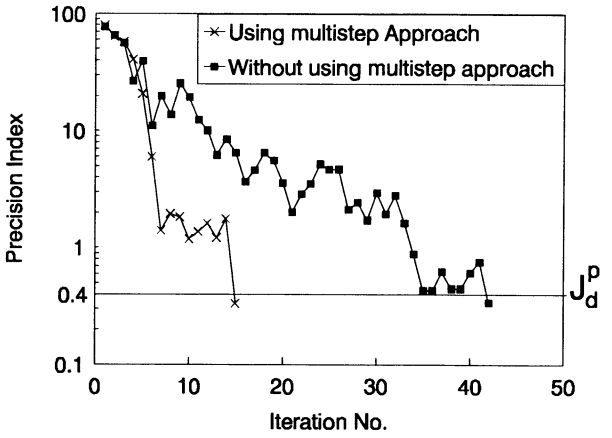


Figure 7: Precision index J^{pi} vs iteration for $J_d^p = 0.4^0$.

3.2 Experimental Results for γ_i

It is difficult to separate the recalling factor γ_i and the modification term $\delta u_i(t)$ from the control input of the human operator as given in equation (1). Therefore to get an idea about the recalling factor γ_i associated with the iterative learning of the human operator, normalized cross correlation coefficient $\mu(i)$ between the control input $u_i(t)$ of i^{th} iteration and the control input $u_{i-1}(t)$ of $i-1^{th}$ iteration is calculated. The normalized cross correlation coefficient $\mu(i)$ is

$$\mu(i) = \frac{1}{N} \frac{\sum_j (u_i(j) - \overline{u}_i)(u_{i-1}(j) - \overline{u}_{i-1})}{\sigma_{u_i} \sigma_{u_{i-1}}} \quad (6)$$

where $\overline{u}_i$, $\overline{u}_{i-1}$, σ_{u_i} and $\sigma_{u_{i-1}}$ are the mean value of u_i, the mean value of u_{i-1}, the standard deviation of u_i and the standard deviation of u_{i-1} respectively. N is the size of vectors u_i and u_{i-1}.

It will give us some clues about how the human operator improves his recalling factor γ_i during the iterative learning. The cross correlation coefficient between the control input $u_i(t)$ and the control input $u_{i-1}(t)$ is plotted in Figure 6[9]. An exponential function is fitted to the data and the fitted cross correlation coefficient $\widetilde{\mu}(i)$ for i^{th} iteration is found to be

$$\widetilde{\mu}(i) = 0.9497(1 - 0.7468e^{-\frac{i}{21.35}}) \quad (7)$$

It suggests the steady state value as 0.9497, which is very obvious as it is almost impossible for a human operator to replicate his control action exactly 100%.

4 Experiments relating the modification term $\delta u_i(t)$

To study the modification term $\delta u_i(t)$, same experimental setup as described in section 2.2 is used. The information display is also modified to overcome the difficulty of the human operator in performing high precision tracking task due to the visual information perception error. The ability of the human operator to generate minimum control steps is also improved by varying the gain of the joystick in the iterative learning process. Hence a multi-step approach is adopted which is illustrated as follows: The desired precision index J_d^p is defined prior to the experiment. The experiment is categorized into three steps. In each step, a sub desired precision index is defined, leading to the final desired precision index in the third step. In the first step, the subject is asked to attain a relatively relaxed precision index i.e. $20 \times J_d^p$, which is assumed to be comfortable for the subject. In the next step, the information displayed to the subject is magnified by a factor of five ($\times 5$) to increase the visual perception of the subject and the sensitivity of the joystick is reduced accordingly by five times to give the subject more precise control. Thus a relatively tough precision index $4 \times J_d^p$ is required to be attained. In the third step, display magnification is further increased by four times, resulting in the total magnification of $\times 20$ and the sensitivity of joystick is reduced accordingly. The subject has to attain the actual desired precision index J_d^p.

The task is defined as the controlled maneuvering of the pendulum to the desired final position θ_f with

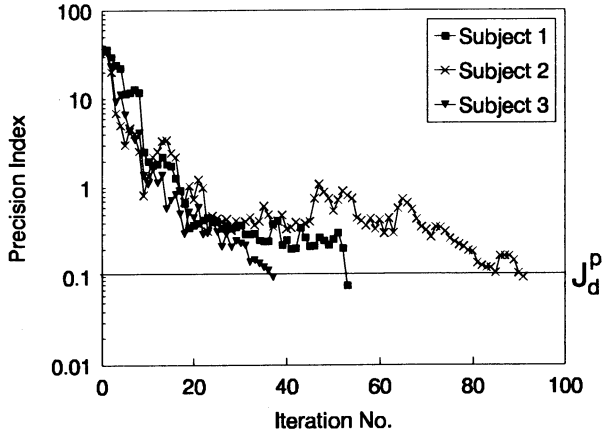


Figure 8: Precision index J^{pi} for $J_d^p = 0.1^0$.

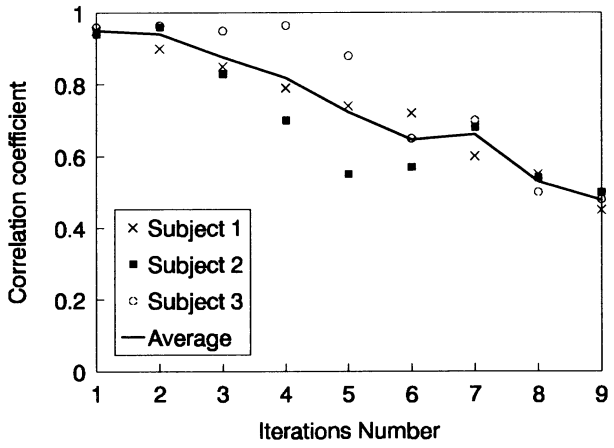


Figure 9: Correlation coefficient of $\delta u(t)$ and $e(t)$.

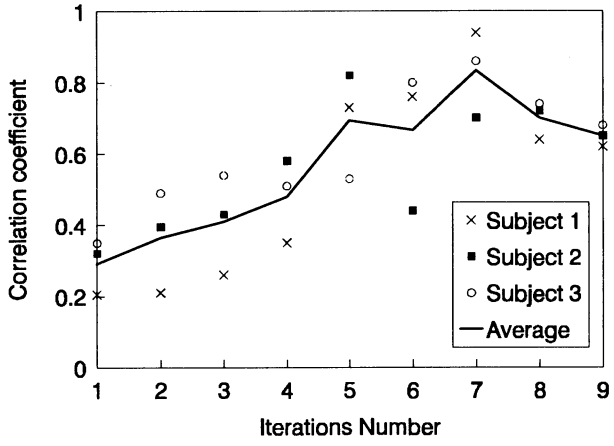


Figure 10: Correlation coefficient of $\delta u(t)$ and $\dot{e}(t)$.

desired precision J_d^p defined in equation (4). To illustrate the advantage of the multi-step information display setup with variable joystick gain control another experiment is performed in which the desired trajectory is same as shown in Figure 2 and the desired precision index J_d^p is set to 0.4^0. In Figure 7, the precision index J^{pi} is plotted on the log scale for the case when normal display is used for the perception of error data by the human operator and the case in which the proposed multi-step information display magnification technique is used. It is clear from the Figure 7 that the human operator can achieve the desired precision index J_d^p very smoothly with fast convergence in about 15 iterations by using the specified technique, hence showing the effectiveness of the multi-step information display magnification technique.

In Figure 8, the precision index J^{pi} of the three subjects is plotted on the log scale for the case when the desired precision index $J_d^p = 0.1^0$. The base-line at 0.1^0 in Figure 8 shows the desired precision index J^{pi}. Although the number of iterations required to achieve the desired precision index is different for the three subjects but the important thing to be noted is that all the subjects have succeeded in attaining the desired accuracy.

5 Analysis of Experimental Results

The above experiments show the ability of the human operator in high precision tracking task. Since the human operator generates the control input based on the information of the perception of the error data displayed to him, therefore for each iteration, error $e_i(t)$, error rate $\dot{e}_i(t)$ and $\delta u_i(t)$ are analyzed to study the human control. It is found that the human operator generates $\delta u_i(t)$ depending on both the error and the error rate. $\delta u_i(t)$ can be represented as

$$\delta u_i(t) = \Lambda_{i-1} e_{i-1}(t) + \Gamma_{i-1} \dot{e}_{i-1}(t) \qquad (8)$$

where Λ_{i-1} and Γ_{i-1} are the learning gains relating to the error and error rate respectively.

One typical example is presented here in which the human operator has to achieve the desired precision index J_d^p equal to 0.4^0 in tracking the trajectory leading to the final desired angle of 90^0. The correlation of the $\delta u_i(t)$ in each iteration is investigated with respect to the error $e_{i-1}(t)$ and the error rate $\dot{e}_{i-1}(t)$.

In Figure 9, the cross correlation coefficient between $\delta u_i(t)$ and $e_{i-1}(t)$ is plotted, whereas in Figure 10, cross correlation coefficient between $\delta u_i(t)$ and $\dot{e}_{i-1}(t)$ is plotted. Results of three subjects are plotted as points and the average of the three data points is plotted as line. It can be observed from the Figures 9 and 10 that the human operator assigns different weights to the error and the error rate data and modifies the gains in each iteration. In the beginning of the experiments when the error is large, the human operator's control is much more like Proportional type iterative learning control ($\delta u(t)$ is a function of the error $e(t)$ only), whereas towards the end of experiment when the error is small but the error rate is large, the human operator's control is similar to Derivative type iterative learning control ($\delta u(t)$ is a function of the error rate $\dot{e}(t)$ only). Moreover, based on the analysis of the data, it suggests that the human operator makes certain rules to handle the task. One simple rule that can be extracted from the above analysis is given as follows: The human operator generates $\delta u(t)$ using the error $e(t)$, when $e(t)$ is prominent, and generates $\delta u(t)$ using the error rate $\dot{e}(t)$ when $\dot{e}(t)$ is prominent.

6 Conclusions

Iterative learning is the common practice utilized by humans. While performing routine tasks, he improves his ability to perform certain task after each iteration according to his past experience. In this paper, the ability of the human operator in the iterative tracking task is investigated. The experiments show the ability of the human operator to reduce the error and to improve the precision index efficiently despite of the unavailability of the plant information. It is also shown that if the perception of the data by the human operator is improved and a better control sensitivity is provided, the human operator can perform the tracking tasks with higher precision. It is also concluded that the human operator performs simple rule based control and modifies the control input in each iteration giving different weights to the error and the error rate data. The human operator shows variable gain PD (Proportional and Derivative) type iterative learning control as given in equation (8) and modifies the weights in each iteration according to some simple rule.

References

[1] Edward W. Vinje and T. P. Edward, "Human Operator dynamics for aural compensatory tracking", *In IEEE Trans. on System, Man and Cybernetics*, Vol. SMC-2 No. 4, 1972.

[2] M. Sasaki, H.Inooka and T. Ishikura, "Manual control of a flexible arm and its application to an automatic control system", *In IEEE Trans. on System, Man and Cybernetics*, Vol. 20 No. 3, 1990.

[3] L. R. Young, "On adaptive manual control", *IEEE Trans. on Man Machine System*, Vol. 10, pp. 292-331, 1969.

[4] H. Inooka and T. Koitabashi, "Experimental studies of manual optimization in control tasks", *IEEE Control System Magazine*, Vol. 10 No. 5, pp. 20-23, 1990.

[5] W. S. Kim, F. Tendick, S. R. Ellis and L. W. Stark, "A comparison of position and rate control for telemanipulation with consideration of manipulator system dynamics", *IEEE J. of Robotics and Automation*, Vol. 3, pp. 426-436, 1987.

[6] R. Ikeura, H. Inooka, "Manual control approach to the teaching of a robot task", *In IEEE Trans. on System, Man and Cybernetics*, Vol. 24 No. 9, pp. 1339-1346, 1994.

[7] M. Arif and H. Inooka, "Investigation of iterative learning control by human operator", *In IEEE conference on System, Man and Cybernetics,* pp:2626-2631, 1997.

[8] M. Arif and H. Inooka, "Experimental study of human performance in high precision tracking task using iterative learning technique", *Interdisciplinary Information Sciences*, Vol. 5(1) pp. 73-85, 1999.

[9] M. Arif and H. Inooka,"Iterative manual control model of human operator", *Biological Cybernetics*, to be published in 1999.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Emergence of Optimal Gait Pattern of Insect in the Real World Attaining its Required Direction and Velocity

Mitsuru SATO, Kazushi AKIMOTO, Yoshinari MAKINO and Masafumi YANO
Research Institute of Electrical Communication, Tohoku University
2-1-1 Katahira Aoba-Ku, Sendai, 980-8577, Japan
E-mail: masafumi@riec.tohoku.ac.jp

Summary It is one of the goals of robotics to realize the autonomous walking robot that can adapt itself to unpredictable changes of environment in the real world. In order to always attain the purpose under unpredictably changing environment, a robot is required to solve the inverse problem. In case of straight walking of insect robot, we have already proposed a new method to solve the inverse problem in which the constraints imposed on the robot are contented by optimally integrating each objective function of the elements through competition and cooperation among them. The insect robot emerges optimal gait patterns attaining a required walking velocity depending on the load. However, when dual purposes, such as walking velocity and direction, are imposed on the robot, the constraints to solve the inverse problem can not be uniquely derived from the purposes. In this case, it is necessary to determine the order of priority of the purposes. If not, there might be competition among the constraints derived from each purpose. Here we propose a new mechanism to solve the ill-posed problem in real time when the system is imposed the dual purposes under the unpredictably changing environment. These mechanisms are quite new and applicable to other artificial control systems.

Key words; emergence, gait pattern, self-referetial, inverse problem, ill-posed problem, insect robot

1.Introduction

It is important for robotics to realize the autonomous walking robot that can adapt itself to unpredictable changes of environment in the real world. In order to control a robot achieving a purpose, it is necessary to solve the inverse problem. Generally, the inverse problem becomes a kind of the ill-posed problem. Usually, this problem has been solved by adding some constraints externally until to make it well-posed problem. In the real world, however, it is impossible to define such an objective function and to set constraints in advance, because the environment changes unpredictably. It means that the traditional control method based on the causality can not apply to the robot attaining the aimed state in the real world.

On the contrary, the motion control system of animals seems to autonomously emerge appropriate information depending on the purposes imposed on the system and unpredictable environment. The walking pattern may be generated through three subregions, the brain, the central pattern generator and the effecter organs. It is remarkable that the efficiency of the energy transduction is optimized at any time by changing its walking patterns. These flexibility are generated by the neural network as a control system, which should be able to organize its dynamical patterns quickly in response to the changes of the walking velocities and load (1-6).

In order to coordinate the movements of the muscles in response to the unpredictably changing environments, the control system should be indefinite. Indefinite system means that the properties of the components of the system are not specified in advance. So in the case of dynamical system, the parameters of the equation are determined to attain the purpose under a given environment unpredictably changed. Because if the control system is definite, it can not adapt to the complexity of the environment. The typical example of the indefinite control system can be seen in central pattern generators termed the polymorphic circuits or multi-functional circuit, but the emergent mechanism of the various spatio-temporal patterns still remains unclear. Recently, we have proposed a new mechanism to emerge various spatio-temporal patterns in polymorphic circuit using a hard wired model (7).

In case of straight walking of insect robot, we have already proposed a new method to solve the inverse

problem in which the constraints imposed on the robot are contented by optimally integrating each objective function of the elements through competition and cooperation among them. The insect robot emerges optimal gait patterns attaining a required walking velocity depending on the load. In this model, the neural system composed of active elements such as nonlinear oscillating elements could self-organize the walking patterns, in which the walking patterns were generated by changing the strength of the interneural connectivities rather than by changing the properties of the composed neurons. So it slightly lacks flexibility in emergence of walking patterns, but the rule for self-organization still seems effective. The walking patterns of the animals are expressed by the coordination of the movement of the effecter organs, that is, muscles, so the mechanism of the emergence of the patterns reflects the characteristics of the muscle contraction. Since muscle have an optimal shortening velocity to provide the highest efficiency, we introduce "the least dissatisfaction for the greatest number of the elements" rule to generate the walking patterns. This rule bring forth the cooperation and/or competition among leg movements, resulting in emerging the most efficient walking pattern (8-11).

However, when dual purposes, such as required walking velocity and direction, are imposed on the robot, the constraints to solve the inverse problem can not be uniquely derived from the purposes. In this case, it is necessary to determine the order of priority of the purposes. If not, there might be competition among the constraints derived from each purpose, resulted in a conflicted states.

So, we extend our method to control the motor systems when dual purposes are imposed on the system under an indefinite environment.

II. Model

a. Characteristics of animal walking

The walking patterns of animals are crucially depending on their walking velocities and the load on them. Since the changes of their surroundings are usually unpredictable, animals should produce walking patterns flexibly to adapt the surroundings. Since the animals change their walking patterns without the brains, walking patterns is not generated by the central nervous system but by the subsystem, below the spinal cord of the mammals and the thoracic ganglions of the insects (3,4). That is, the higher nervous systems are not necessary for the organization of the walking pattern. In the case of stick insect, at high speed the front leg and the hind move simultaneously and the middle antiphasic to the others, forming a tripod to support their body. On the contrary, when they walk slowly, the three legs of each side move metachronally. A pair of legs of the same segment step alternately. As increasing the walking velocity, the insect changes the patterns critically depending on their velocity, resembled to a phase transition. The walking patterns also vary with the load [1,5,6].

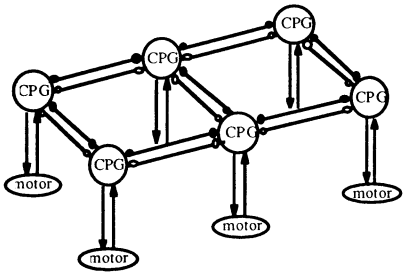
b. Central pattern generator as a control system

The walking of the insect is controlled by the three thoracic ganglions, prothoracic, mesothoracic and methathoracic ganglions (12). These ganglions send motor outputs to control leg muscles and receive the external afferents as to position, load and force of each muscle. These ganglions are interconnected each other through a pair of thoracic connectivities It has been clarified that the well coordinated motion among the legs is organized not only by the neural system composed of the three ganglion, connected through the intersegmental connectivities, but by the mechanical interaction through the movements of legs (12). Central pattern generators (CPGs) are networks of neurons to control the motor system generating spatio-temporal pattern of neural activities.

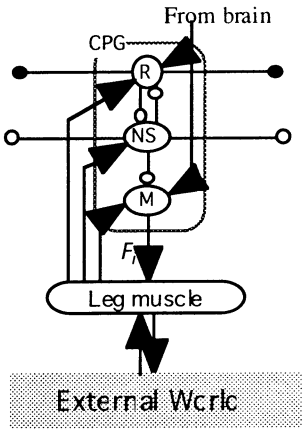
Two kinds of mechanisms for the emergence of polymorphism in the network are possible. One is by changing the strength of the interneural connectivities and the other is by changing the dynamical properties of the neurons composing the network without changes of the connections. The former is widely accepted method to change the characteristics of the network because of its simplicity. In this case, however, it is too simple to generate the complexity of the polymorphism of the network. In this paper, we construct a coupled nonlinear-oscillator system as the polymorphic network. Since, it is well known that an isolated neuron changes its dynamical properties by neuromodulaters, we adopt the latter mechanism to produce polymorphism without changes of the connectivities.

Intersegmental interneurons in a thoracic ganglion of locust has been extensively investigated by Laurent and Burrows (13,14). We adopt fundamentally their results as schematically shown in Fig.1. When insect walks, a motion planning such as walking velocity and direction is expressed by a small number of degrees of freedom, which is first made at higher center such as brain. The output

made in brain according to this motion plan is sent to the three thoracic ganglions. The output from the higher center can be regarded as a feed forward signal of this system and is given a constant (DC input) for simplicity. In thoracic ganglion, this signal is transformed into rhythmic wave by rhythmic neuron corresponding to a spiking interneuron in the ganglion. The rhythmic neuron makes direct synaptic connection with nonspiking inter-neuron (NS neuron) which is great important to integrate the information on the states of muscles and intersegmental path way. NS neuron



A



B

Fig.1 A; Intersegmental connection among CPGs. B; Structure of each CPG which controls leg muscles. R, NS and M denote rhythmic neuron, non-spiking interneuron and motor neuron, respectively.

transforms the output of the rhythmic neuron to send the motor neuron. Intersegmental connection between rhythmic neurons in neighboring CPGs are inhibitive, which produce asynchronous oscillation between neighboring rhythmic neurons. The frequency of the rhythmic neuron determine the walking velocity and the inhibitory interaction among them enables to appear any phase relationship among the movement of legs. In this sense, the rhythmic neuron is a kind of command neuron which receives the information of walking velocity, that is, purpose of the animal created in the brain, but the spatio-temporal pattern of the movement of legs are emerged by integration of the dynamical information of the effecter organs in the NS neurons under the constraint of the purpose.

We use the KYS oscillator as the neuron model for rhythmic neuron as follows;

$$\dot{x}_R=-y_R-(A_{R1}x_R^3+B_{R1}x_R^2+C_{R1}x_R)$$

$$\dot{y}_R=(A_{R2}x_R^2+B_{R2}x_R+C_{R2})x_R+D_R \qquad (1).$$

where x denotes voltage of neuron and Dr is the input to the rhythmic neuron which determine the frequency of the oscillation. The NS neurons is given by eqn. (2).

$$\dot{x}_{NS}=-y_{NS}-(A_{NS1}x_{NS}^3+B_{NS1}x_{NS}^2+C_{NS1}x_{NS})+\beta_R x_{Ri}$$
$$\dot{y}_{NS}=(A_{NS2}x_{NS}^2+B_{NS2}x_{NS}+C_{NS2})x_{NS}-D_{NS}+\sum_j \alpha_{NSij}x_{NSj} \qquad (2)$$

where Dns is the input from the rhythmic neuron which controls the phase relationship among the movement of legs. And the motoneuron is governed by the following equation.

$$F_i=G_m sigmoid\,(x_{NS}) \qquad (3)$$

where F_i is the activity of the motoneuron which determines the motive force of the leg.

c. Sensory feedback

In order to self-organize the walking pattern according to the circumstance, it is necessary to obtain the information on the surroundings and the state of the legs. At the beginning of the stance phase, only the posterior muscle shortens, but at the end of the stance phase the position sensor of the posterior muscle should strongly inhibit the motoneuron of it, activating the motoneuron of the anterior muscle. In the case of the swing phase, the interaction between the pair of muscles should be reversed. These interactions can be presented by the direct synaptic connection of the position sensor of each muscle with the motoneurons and by the feedback to the connectivities between the nonspiking neuron and the motoneuron as shown in Fig. 1. The hind leg moves antiphasic to the middle, which also moves antiphasic to the front leg, although there is no strong coupling between the hind and the front legs. So, the information required to optimize the efficiency of energy transduction is given as follows; 1) the feedback information from leg to motoneuron is

$$\Delta G_{mi} = k_{\eta}\left(\frac{\partial \eta_i}{\partial F_i} - \frac{\sum_{i \neq j} F_j \frac{\partial \eta_j}{\partial F_j}}{\sum F_j}\right) \quad (4).$$

It means that the legs moved synchronously tend to share the load equivalently, where η denote the efficiency curve of the energy transduction of muscle. Each leg requires to work more efficiently, so the feedback to NS neuron is

$$\Delta D_{NS} = k_{Th} \frac{F_i \frac{\partial}{\partial}}{\nabla} \quad (5).$$

This feedback information determine the degree of the synchronization among the legs.

d. Turning motion

One of the aims of biological motion is to reach its destination. If dual purposes such as a required walking velocity and a destination are imposed on the system, the arrival to the destination takes the priority over all other purposes. The velocity of stance phase of right and left side legs is tuned proportional to the angle θ between the axis of the body and the direction of the destination, which is feed backed until $\ddot{\theta}$ (second derivative of θ) becomes zero. If the $\dot{\theta}$ times an estimated time do not satisfy the required angle θ , the strength of the feedback should be enhanced.

III. Results

a. Straight Walking

In case of straight walking, the required velocity is the only purpose of the robot, which is the strong constraint for our model system to attain at any required velocity and any load on the system. Our insect robot can fundamentally generates the two different walking patterns depending on the walking velocities and loads. The walking patterns are characterized by the phase relationship among the six legs, showing the walking pattern of metachronal gait. The phase relationship between the hind and the front drastically changes as the walking velocity increases. As increases the velocity, our robot shows that the front and the hind legs move simultaneously, called tripod gait as reported previously.

b. Turning walk

In this model, the structure of leg is composed of only two muscles, flexer and extenser muscles, so the movements of legs are limited to move parallel to the axis of the body. When the angle θ between the axis of the body and the direction of the destination is large, the walking velocity should become slower and the gait pattern is metachronal. When θ is small, the insect can turn at higher velocity with a tripod gait. At intermediate angle, outer side legs and inner side legs take tripod and metachronal gait, respectively.

V. Discussions

We have simulated an insect robot as an example that can generate appropriate walking patterns to walk efficiently. Since the walking pattern changes crucially depending on their walking velocities and loads, animals could generate a great number of diversities of walking patterns to adapt the unpredictable changes of their surroundings.

We have also showed that a new control mechanism installed in the insect robot, which can walk attaining the dual purposes of the system as possible as it can operate at higher efficiency of energy transduction under unpredictable changes of the environment. This control mechanism is derived from a metarule to determine the priority among the purposes. In case of turning walk, the destination takes the priority over all other purposes.

References

(1) Pearson, K.G., (1972). J.Exp.Biol. 56:173-193

(2) Pearson, K.G, (1976). Sci. Am. 235, 72-86

(3) Graham, D., (1979a). Biol. Cybern. 32:139-145

(4) Graham, D., (1979b). Biol. Cybern. 32,147-152

(5) Foth E., Graham D. (1983a). Biol. Cybern. 47:17-23

(6) Foth E., Graham D. (1983a). Biol. Cybern. 48:149-157

(7) Makino Y., Akiyama M. & Yano M., (1999). Biol. Cybern. (in press)

(8) Kimura S., Yano M., & Shimizu, H., (1993). Biol. Cybern. 69 183-193

(9) Kimura S., Yano M., & Shimizu, H., (1994). Biol. Cybern. 70 505-512

(10) Akimoto K., Watanabe S., Makino, Y. and Yano, M. (1999). IEEE Systems, Man and Cybernetics, Proceedings VI-942-947

(11) Akimoto K., Watanabe S., Makino, Y. and Yano, M. (1999). Artificial Life and Robot vol.3, 102-105

(12) Dean, J., (1989). J. Exp. Biol. 145, 103-131

(13) Laurent, G., & Burrows, M., (1989a). J. Neurosci. 8, 3019-3029

(14) Laurent, G., & Burrows, M., (1989b). J. Neurosci. 8, 3030-3039

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Breakthrough for Artificial Heart Control Based on Adaptability of Circulatory System

Makoto Yoshizawa*[1)], Akira Tanaka[2)], Keni-chi Abe[1)],
Hiroshi Takeda[3)], Tomoyuki Yambe[2)], Shin-ichi Nitta[2)]
*E-mail: yoshi@abe.ecei.tohoku.ac.jp
1) Graduate School of Engineering, Tohoku University, Sendai 980-8579, Japan,
2) Institute of Development, Aging and Cancer, Tohoku University, Sendai 980-8579, Japan,
3) Faculty of Engineering, Tohoku Gakuin University, Tagajo 985, Japan

Abstract

One of the most substantial problems in controlling the total artificial heart (TAH) is that no one knows what amount of cardiac output should be pumped out from the TAH to the systemic circulation. To cope with this problem, the peripheral vascular resistance-based control method has been proposed and attained a desirable success for long term survival in animal experiments. In this study, it has been theoretically shown that the method relies on adaptability of the cardiovascular center: capability of changing its own control algorithm according to a new environment caused by introduction of the TAH.
Keywords: artificial heart, peripheral vascular resistance-based control, adaptive control, cardiac output.

1 Introduction

One of the most important problems in controlling the total artificial heart (TAH) is how to determine the quantity of the cardiac output, i.e., the flow pumped out from the TAH to the vascular system [1]-[7]. Although many methods for solving this problem have been proposed so far, any definite methods have not yet been found. In this situation, the so-called *1/R control method* proposed by Y. Abe [8], [9] has been highly evaluated because the method could attain the longest survival record (532 days) in the animal experiment with a perfectly automatic controller.

The 1/R method is categorized into the *peripheral vascular resistance(PVR)-based control method* in which the cardiac output is automatically determined according to the value of the PVR. In this study, it has been theoretically shown that the PVR-based control method relies on adaptability of the blood circulatory system: capability of changing its own control algorithm according to a new environment caused by introduction of the TAH.

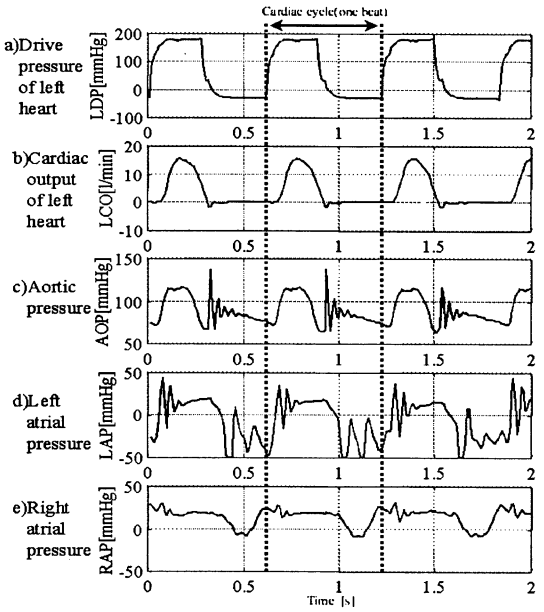


Fig. 1: An example of waveforms generated by an air-driven TAH.

2 Model of Circulatory System Before and After Implantation of the TAH

2.1 Two-Input Two-Output Model

A typical example of waveforms generated by an air-driven TAH implanted to an adult goat is shown

as Fig.1. The drive pressure a)LDP was provided for the TAH to pump out blood, and then cardiac output of the left heart b)LCO, the aortic pressure c)AOP, left atrial pressure d)LAP and right atrial pressure e)LAP were generated.

Even if the natural heart is replaced by the TAH, the cardiovascular center is still aware of the optimal quantity of the cardiac output. The PVR-based control method aims at utilizing this fact.

First, consider a considerably simplified electrical circuit model of the circulatory system before implantation of the TAH as shown in Fig.1. The cardiac output CO pumped out by the natural heart is temporally stored in the compliance C_{AOP}[L/mmHg] of the aorta. The averaged cardiac output $\overline{CO}$[L/min] goes through the PVR R[mmHgmin/L] to the right atrial compliance C_{RAP}[L/mmHg], and finally reaches to the pulmonary circulation.

The peripheral vessels contract and relax under control of the cardiovascular center via the autonomic nerve. This action corresponds to the manipulation of R. The change in R directly causes the change in the aortic pressure AOP. Simultaneously, $\overline{CO}$ is also changed by changing in R. On the other hand, the manipulation of CO via stroke volume and heart rate changes not only $\overline{CO}$ but also AOP. This means that the two-input two-output system $(CO, R) \rightarrow (\overline{CO}, AOP)$ is a coupled system. Thus, it can be guessed that the cardiovascular center must manipulate R and CO as a decoupling controller so that $\overline{CO}$ and AOP can independently approach their corresponding reference values $\overline{CO}^*$ and AOP^*.

2.2 Decoupling Control and Adaptation of Cardiovascular Center

Next, consider the case where the natural heart is replaced by the TAH as shown in Fig.2. In this situation, it is a matter of course that the cardiovascular center cannot control the TAH. On the contrary, the controller of the artificial heart cannot know the reference value $\overline{CO}^*$. Thus, it is impossible for the TAH to control its outflow according to the true physiological demand.

However, the cardiovascular center can still know its own true reference value of $\overline{CO}^*$ and monitor the difference between $\overline{CO}^*$ and the actual $\overline{CO}$. Hence, the following control strategy can be proposed.

In the new control strategy, the cardiovascular center is forced to manipulate R to make $\overline{CO}$ approach

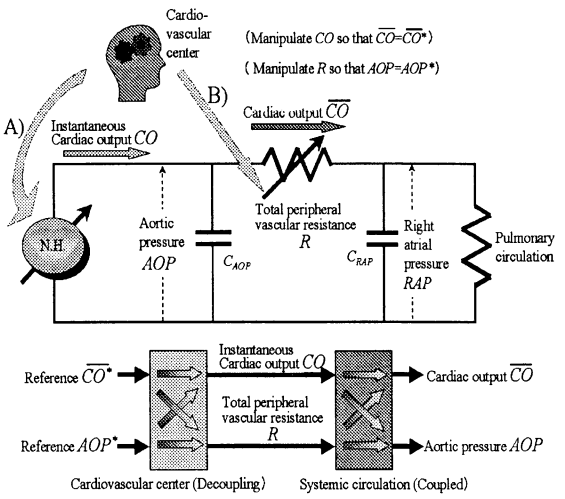


Fig. 2: Decoupling control behavior of the cardiovascular center before implantation of a TAH.

$\overline{CO}^*$, which means flow control via the PVR. On the contrary, the TAH controller is force to manipulate CO to make AOP approach AOP^*, which means pressure control. However, the pressure control needs the true reference value of AOP^* which is also unknown, and thus a tentative reference value AOP^{**} is used instead of AOP^*.

To guarantee the feasibility of the control strategy, the following three conditions are required.

(a) The tentative AOP^{**} is close to the true AOP^*.

(b) The change in R does not affect the pressure AOP but the flow CO.

(c) The cardiovascular center has an adaptability to change its own control strategy from the pressure controller to the flow controller.

Of course, it is difficult to make AOP^{**} close to AOP^* but the possible dynamic range of AOP^* is considerably narrower than that of the reference cardiac output CO^*. Hence, it can be expected that the effect of the setting error with respect to AOP^* on the recipient's hemodynamics may be smaller, and the condition (a) may be satisfied more easily in comparison with CO^*.

The condition (b) will be realized if the TAH controller has a decoupling control mechanism to suppress the effect of R on AOP. The value of R can be ob-

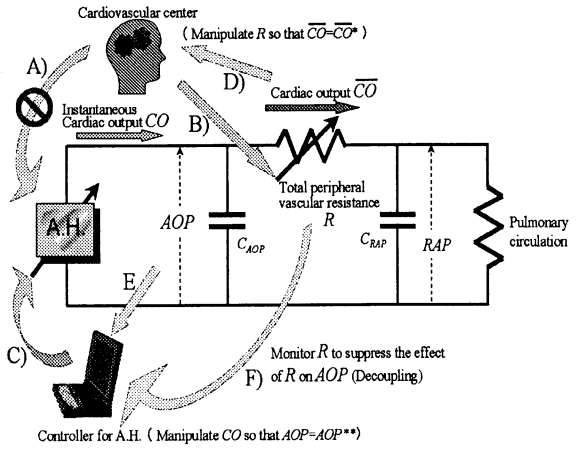


Fig. 3: Cooperation between the cardiovascular center and the TAH controller after implantation of the TAH.

tained as

$$R = \frac{AOP - RAP}{CO}, \tag{1}$$

where RAP is the right atrial pressure. Because $\overline{CO}$ cannot be observed, CO is used as the denominator in (1) instead of $\overline{CO}$. The condition (b) allows the cardiovascular center to manipulate R for the sake of the flow control because the change in R can directly affect the change in flow.

If the condition (b) is satisfied, the possibility that the condition (c) can hold will increase. The most important problem is whether the cardiovascular center has such a drastic adaptability as the condition (c).

3 PVR-Based Control

3.1 1/R Control Method

The 1/R control method proposed by Y. Abe [8] is a special case of the PVR-based control methods. In this method, the reference cardiac output CO^* is given by a quite simple equation:

$$CO^* = aR^{-1} + b(AOP - AOP^{**}), \tag{2}$$

where a and b are constants. This equation means that CO^* is determined by a linear combination of the term proportional to the reciprocal of R and the term of difference between AOP and AOP^{**}. The former term operates for generating CO depending on the PVR and the latter term functions for regulating AOP. The 1/R control method could attain the longest survival (532 days) of an animal implanted with a TAH which has never been updated yet. Currently, the 1/R control method has been well evaluated because the method could yield a good hemodynamic state with a low central venous pressure while other automatic control methods for the TAH have a tendency to increase the pressure.

The deficiencies of the 1/R control method are as follows: 1) there is no theoretical method for determining the coefficients a and b in (2), which means that a lot of experiments with trial and error are required; 2) the decoupling control characteristic corresponding to the condition (b) is weak.

3.2 PVR-Based Adaptive Control

To cope with these problems, the authors have introduced an adaptive controller, and attained successful results[10].

First, a deterministic autoregressive moving average (DARMA) model was used to approximate the dynamic characteristic of the two-input one-output system $(CO, R) \rightarrow AOP$. Identifying the coefficient parameters of the DARMA model beat by beat, the TAH controller manipulates CO so that AOP can approach AOP^{**} and cancel the effect of R on AOP. As a result, the TAH controller functions for the decoupling adaptive control which yields the reference value CO^* represented by

$$\begin{aligned} CO^* &= f(R, AOP, AOP^{**}) \\ &= f(CO, RAP, AOP, AOP^{**}). \end{aligned} \tag{3}$$

The key point of this algorithm is to arrange an environment in which the intention of the cardiovascular center may not appear in the change in AOP but directly appear in CO^* by way of R. Because the function $f(\cdot)$ is an adaptive controller, fine tuning of its parameters is not required. Thus, it can be expected that the new controller can overcome the deficiencies of the 1/R control method.

Fig.3 shows an example of control behavior of 1/R control method and the PVR-based adaptive control method for an air-driven TAH implanted to an adult goat. At the interval of the middle part of the abscissa, the goat was forced to walk on a treadmill.

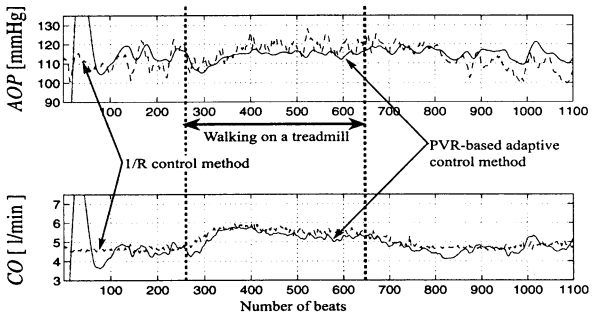


Fig. 4: Control behavior of 1/R control method and the PVR-based adaptive control method for the TAH.

Figs.3 a) and b) are the responses of AOP and CO of the TAH with the number of beat, respectively. It can be seen that the PVR-based adaptive control method (solid line) has similar waveforms as 1/R control method (broken line). The figure shows that after the goat began to walk, CO increased automatically while AOP remained roughly constant. This means that the cardiovascular center could adjust the cardiac output according to its own physiological demand.

The practical problem of the PVR-based control methods is that it is quite difficult to measure AOP and RAP included in (3) with sufficient accuracy during a long term inside the body [5].

4 Conclusion

As mentioned above, the PVR-based control method has a possibility to give a breakthrough for the cardiac output control problem of the TAH because the method can avoid directly finding the true physiological demand of the cardiac output. However, the theoretical adequacy of the method cannot be ascertained only from the viewpoint of control theory. This is because the method relies on the adaptability of the cardiovascular center which has not sufficiently been identified yet, and then it is not perfectly predictable.

References

[1] Maslen EH, Bearnson GB, *et al.* (1999), Feedback control applications in artificial hearts. IEEE Control Systems 18(6): 26-34

[2] Allen G, Murry K and Olsen D (1996), Control of the artificial heart. ASAIO J. 42: 932-937

[3] Hiller K, Seidel W and Kolff W(1962), Servo mechanism to drive an artificial heart inside the chest. Trans ASAIO 8: 125-130

[4] Kaufman R, Becker K, *et al.* (1995), Fuzzy control concept for a total artificial heart. Artificial Organs 19: 355-361

[5] Nakamura M, *et al.* (1999), The development of a control method for a total artificial herat using mixed venous oxygen saturation. Artificial Organs (in press)

[6] Kitamura T (1996), Structural analysis of a circulatory system for artificial heart control based on the Coleman's model HUMAN, Proc Waseda Intl Conf Modeling and Simulation Technology for Artificial Organs, Tokyo: 107-108

[7] Tsach U, Geselowitz DB, *et al.* (1989), A novel output feedback pusher plate controller for the Penn State electric ventricular assist device. J Dynamic Systems Measurement and Control 111: 69-71

[8] Abe Y, Chizei T, *et al.* (1997), Long-term hemodynamics and pathophysiology in a total artificial heart goat survived for 532 days with 1/R control. Japanese J Artificial Organs 26: 21-26 (in Japanese)

[9] Abe Y, Chizei T, *et al.* (1998), Physiological control of a total artificial heart: conductance- and arterial pressure-based control. J Applied Physiology 84(3): 868-876

[10] A. Tanaka, M. Yoshizawa, *et al.* (1997), Experimental analysis of identification function and stability of an adaptive controller for total artificial heart. Japanese J Artificial Organs 26: 114-118 (in Japanese)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Collective Behavior of Multi-agent System with Simple Interaction

K. Sugawara, M. Sano*, I. Yoshihara†, K. Abe‡, T. Watanabe

Graduate School of Information Systems, The Univ. of Electro-Communications.
Chofu, Tokyo, 182-8585 JAPAN
*Research Institute of Electrical Communication, Tohoku Univ.
Sendai, Miyagi, 980-8577 JAPAN
†Dept. Computer Science, Miyazaki Univ.
Miyazaki, Miyazaki, 889-2192 JAPAN
‡Graduate School of Engineering, Tohoku Univ.
Sendai, Miyagi, 980-8579 JAPAN

E-mail: sugawara@sd.is.uec.ac.jp

Abstract

We investigate the collective behavior of simple multi-robot system. In this paper, After defining the concept of swarm intelligence, we examine collective behavior of multi-robot system through a task to gather pucks in the field. An architecture of the robot assumed in this paper is very simple: it has a drive system and the simplest means of interaction. Measuring the effectiveness of group behavior by task completion time or the number of collected pucks, we discuss emergence of swarm intelligence from a viewpoint of quantitative aspect.

1 Introduction

Social insects such as ants and bees establish well-ordered societies even in the absence of particular individual intelligence[1]. This is because functions are self-organized in their societies through interactions with each other. This fact suggests that we may be able to synthesize adaptive systems just like their societies by coordinating interactions between each element even if the function of each element is not so complex.

In robotics field, many researchers have been studying the multi-robot system[2,3,4]. Most of them gave us qualitative aspect of robots behaviors, but few researches discussed the quantitative aspect of MRS. To make an useful system, it is necessary to investigate the characteristics and the design method of the multi-robot system. It means we have to discuss not only qualitative aspect but also quantitative aspect when we investigate MRS.

In this paper, we mainly focus on the quantitative aspect of the system using simple interacting multi-robots.

2 Multi-robot System and Swarm Intelligence

The behavior emerging from interacting elements is called "swarm intelligence"[2]. There is not strict definition of swarm intelligence, but Beni claimed the meaning of swarm intelligence from the viewpoint of quantitative aspect[5]. N autonomous elements act asynchronously in an environment. Beni's claim is that swarm intelligence emerges from interacting among N elements only when N exceeded a critical number N_C. It means that $W(N)$, which implies the work achieved by interacting units, is described by a function with a critical number N_C, and $W_0(N)$, which implies the work achieved by N independent elements, is described as zero for all N.

A characteristic of swarm intelligence defined by Beni is that the relation between the number of elements N and the amount of useful work W is nonlinear due to the critical number of N_C. This characteristic enables us to generalize the definition of swarm intelligence[6]. Emergence of swarm intelligence is defined as $W(N) > W_0(N)$,*i.e.*, the work performed by N interacting elements is larger than the sum for N individuals.

3 Foraging Behavior of Multi-robots

To examine an emergence of swarm intelligence, we consider collective behavior and evaluate the efficiency of the group through a task to collect pucks distributed in a field using simple interacting multi-robots.

Each robot has a light and a pair of photo sensors. Each robot has three modes: searching, broadcasting and homing.

- Searching: to move toward the signal source.
- Broadcasting: to radiate singal light isotropically.
- Homing: to move toward home.

The robots interact only when a robot radiates a light signal. This duration is called "interaction duration" in this paper.

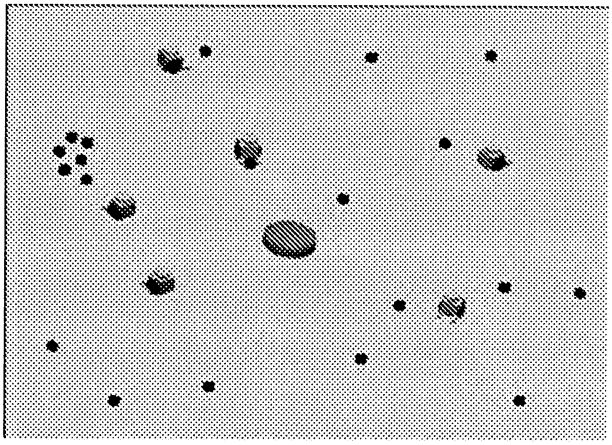


Figure 1: Schematic of foraging task

● Emergence of Beni's Swarm Intelligence

Can we easily realize swarm intelligence defined by Beni ? Let's think about the system in which each unit moves toward feeding point by communicating with each other like ants' society. Let's denote x, y as the number of elements which transport foods and the number of elements whic search foods, respectively. Total amount of elements is N. When a searching element finds a transporting element, its mode changes to transporting mode. Here, we describe the probability as α. A transporting element sometimes loses their way and changes to searching mode. We describe this probability as β. In this case, the master equation of the system can be described as follows:

$$\begin{aligned} \dot{x} &= \alpha xy - \beta x \\ \dot{y} &= -\alpha xy + \beta x \\ N &= x + y. \end{aligned} \tag{1}$$

In steady state, we can easily find out the system shows non-linear relation between the number of elements N and the number of transporting elements. There is a critical value N_C that searching elements appear in this system and this phenomenon corresponds to the swarm intelligence defined by Beni.

● Emergence of Generalized Swarm Intelligence

- Collecting to Unfixed Point

Deneubourg et.al reported food gathering behavior of ants could be described by simple model[7]. According to their model, the ants' behavior is determined stochastically. In case that an ant meets foods frequently, the probability that the ant puts a food is high, if it has a food. On the other hand, the ant finds foods barely, the probability to pick up a food which the ant finds becomes high. In this model, as the ants work with indirect interaction, their performance is in proportion to the number of ants.

What happens when we introduce simple direct-interaction between each robot? Here we assumed the following direct interaction and measured the number of pucks in the largest cluster. Focusing on the relation between the number of units N and the number of pucks P in the cluster, we can obtain the relation

$$P \sim N^{\beta(z)}, \tag{2}$$

where the value of β depends on the interaction duration z.

- Collecting to Fixed Point

In case that each unit collects foods to fixed point, we can consider various types of puck distribution. Introduction of fractal distribution enables us to treat the condition of the field quantitatively. Focusing on the relation between the number of units N and the task completion time T, we can obtain the relation

$$T \sim N^{-\beta(D,z)}, \tag{3}$$

where the value of β depends on the dimension of the puck distribution D and the interaction duration z.

The behavior of robots can be described by state transition diagram and analyze mathematically.

● Emergence of Rotational Behavior

In case that there are some localized puck distributions, we can observe a kind of emergence phenomenon. We assume 8 localized puck accumulations around the nest. Each accumulation equidistant from the nest has 400 pucks and is placed at equal intervals.

When the interaction is weak (that is, short interaction duration or short interaction range), they work in parallel. It means that robots flock around all accumulations uniformly. But when the interaction is strong, we can observe they work in series. It means that all robots flock around an accumulation intensively and when the accumulation is eaten out, they flock around other accumulation. In this case, their behavior is rotational. A probability that such a rotational phenomenon occurs depends on the number of robots and the strength of the interaction.

● Finite Interaction Range and Pattern Formation

An introduction of a finite interaction range and an inactive state enables us to show another interesting behavior. When a robot finds a food, it stays there, and it turns on and off its light for a constant period. Other robots within a interaction range react to this light. They move towards the light source, and when the light goes out, they stop there and react to nothing for a constant period. This algorithm enables the system to form a streaming pattern and to avoid obstacle between the feeding point and their home.

we applied this system to foraging task and measured the effectiveness of the system focusing on the relation between the number of robots and the number of collected pucks per a robot. As a result, we found that the system emerges the generalized swarm intelligence.

4 Discussion and Conclusion

We studied the swarm intelligence through the task to collect pucks distributed in the field. In this study, the definition of swarm intelligence was generalized at first. Next, using simple interacting robots, we showed phase transition-like behavior, generalized swarm intelligent behaviors, and emergence of other behaviors such as rotational phenomenon and pattern formation. In generalized swarm intelligent behaviors, we evaluated the group behavior quantitatively. We examined the value of the power law exponent by computer simulation and found that it changes from -1.8 to 0 by changing the interaction duration. On the other hand, we have already obtained that it changes from -2 to 0.

Now we are studying their behavior using real micro robots system.

References

[1] B. Hölldobler and E. O. Wilson, *JOURNEY TO THE ANTS*, Harvard University Press, (1994).

[2] K. Asama, et al., Eds. *Distributed Autonomous Robotic System 2*, Springer-Verlag, (1996).

[3] T. Balch and R.C. Arkin, "Communication in Reactive Multiagent Robotic Systems," *Autonomous Robots*, 1, (1994), pp. 27.

[4] M.J. Mataric, "Interaction and Intelligent Behavior," Ph.D Thesis, MIT, (1994).

[5] G. Beni and J. Wang, "Distributed Robotic Systems and Swarm Intelligence," *Proc. of IEEE Int. Conf. Rob. Autom.*, April, (1991), pp. 1914.

[6] K. Sugawara and M. Sano, "Cooperative Acceleration of Task Performance: Foraging Behavior of Interacting Multi-Robots System," *Physica D*, 100, (1997), pp.343.

[7] J. L. Deneubourg, S. Goss, N. Franks, A. Sendova-Franks, C. Detrain and L. Chretien, "The Dynamics of Collective Sorting Robot-Like Ants And Ant-Like Robots," *From Animals to Animats*, MIT Press, (1990). pp. 356.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

On the possibility of "complex systems biology"

(extended abstract)

Hiroshi Tanaka
Department of Information Medicine
Medical Research Institute
Tokyo Medical and Dental University, Tokyo, Japan 113-8510

Systems approach to "What Is Life?"

Despite the remarkable advances in the recent molecular biology, specially in the fields of molecular genetics, there has been no sufficiently scientific answers as yet to the question of "What is life? Or "What is the differences between the non-living and living things"?"

We can enummerate several characteristics to describe these differences such as self determination of boundary, metabolism and self-replication, evolution and so on. Seeing the list of these characteristics, we could say that what makes life as it is lies not in the property of substances but in the properties as a system. Differences of substances are not crucial for the distinctions. Thus, we must first denote bellow.

Proposisition 1: Life is a special class of organisation of substances.

But what kind of special orgaization? It determines its own boundary, continously is producing its own components by its chemical reaction network, and has genetic information by which it can make itself. Thus, life as a system always relates to itself. We would say life is self-related system. This property is called self-referential system or autopoetic system.

Proposition2: Life is self-structured system.

But what is self? In what level, "self structure" emerges. If there exist equivalence as to physical feasibility, the physical causality does not work. When it comes to the level where selection among the physically equivalent states, the concept of information merges. Self emerges at the level of information.

In more concrete word, life is a cyclic renewing process in itself. A little different but essential same as this fundamental concep could be found elsewhere, where life is thought as a continous reproduction of (productive) network of its components. As a matter of fact, the componets of life is continously being reproduced by the chemical metabolic reactions, and this reaction network itself is brought its existence through reaction itself.

Self-structured system and Information

But what is the difference of the self structure between biotic and abiotic self-systems? It is the information-embededness for the right response to the enviromental change. The complexity of the abiotic system is restricted,, thought it exhibits seemingly very complicated pattern, whereas in biotic self structure information of enviroment stored in its self structure. Thus, we distinguish the complexity levels of organisations as follows:

(1) Spontaneously emergent complexity
(2) Information mediated complexity

Nonequilibrium self organised system with autocatalytic reaction is considered as the origin of self maintaineous system. We do not think that the biotic self structure continously evolved from the abiotic self organisation. But we conjecture that biotic self structure "phase transition" started when the synbiosis of the information cycle and metabolic cycle occur, or in other words, metabolic recurrent reaction network is taken over bv informational self cycle. On account of this incorporation of information structure into semi-self, the semi-self structure can accumlate the information as to respond the enviroment effectively to form full-self structure.

Proposition 3: Synbiosis of the informational self structure and abiotic nonequilibrium self-organisation is the start of protoform of current life.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A Compact 8-node Linux PC Cluster for Protein Information Analysis

Yutaka Akiyama, Tamotsu Noguchi, Kentaro Onizuka and Makoto Ando
Parallel Application TRC Laboratory, Real World Computing Partnership
Tsukuba Mitsui Bldg. 1-6-1 Takezono, Tsukuba-shi, 305-0032 Japan
Email: {akiyama, noguchi, onizuka, ando}@trc.rwcp.or.jp

Abstract

We describe biological and chemical parallel applications running on our PC clusters: the PAPIA system and the MolTreC no-cutoff molecular dynamics simulator. We have developed and successfully ported those parallel application programs onto a new compact 8-node Linux PC cluster. The PAPIA system is an integrated parallel computing system dedicated to typical search/analysis problems required in protein research, such as structure similarity search, sequence homology search, and multiple sequence alignment. The system was first implemented on a 64-node PC cluster (PAPIA cluster) and we have been operating a WWW-based free service since April 1998. This service has been accessed from 63 countries in the world to date. Now the PAPIA system also runs on a compact and inexpensive 8-node cluster (PAPIA mini cluster).

In addiation, we have developed a parallel molecular dynamics simulation program called MolTreC which employs a parallel Barnes-Hut tree code for calculating long-range Coulomb interactions. The MolTreC program can rapidly calculate the Coulomb interactions of proteins in water without any cutoff approximations. The PC clusters show good scalability in these two real parallel applications.

Keywords: Protein analysis, Genome analysis, PAPIA system, Molecular dynamics, PC clusters

1 PAPIA system

Recently, analysis of protein molecules and genetic DNA sequences has depended heavily on the power of computational approaches. "Bioinformatics" or "Computational biology" has been rising as a new and active area of study which is necessary for understanding disease mechanisms, designing drugs and macromolecular materials, and improving agricultural resources.

Molecular biology databases are rapidly growing in size (a 10 fold increase in 3 years, in the case of the GenBank DNA database) and the entire genomic sequencing of several organisms has or will be finished in a few years. The draft sequences from the human genome project will be available in 2000, and a full version of the human genome sequences will be obtained by 2003. Vast amounts of calculations are required for the analysis of huge databases, but fortunately, large-scale parallelism can be exploited in most cases. We are aiming at demonstrating the power of the latest parallel calculation technology in efficiently solving computational biology problems.

For this purpose, we have implemented the **PAPIA** (**PA**rallel **P**rotein **I**nformation **A**nalysis) system [1]. The PAPIA system performs fast parallel calculations in protein analysis, such as 1) structure similarity search, 2) sequence homology search, and 3) multiple sequence alignment, with high parallel efficiency on a distributed-memory parallel computer.

The systems has been implemented on two PC clusters (the PAPIA cluster [1] and the PAPIA mini cluster, shown in Fig.1 and Tab.1) and also on Hitachi SR2201 massively parallel computer.
The PAPIA system has several unique features[2]:

(1) Written with a C++ class library (called PAPIA library [3]) which was designed to easily and consistently develop protein analysis programs.

(2) Parallel processing on Master-Worker scheme.

(3) Fully on-memory database search is achieved with a daemon-like PAPIA server program.

(4) WWW-based free service has been operated through the Internet.

A single calculation server program (PAPIA server) was designed. Any calculation service can be performed by the same server program with a specific function call. The server program is launched simultaneously on all calculation nodes, at the booting of the PAPIA cluster. The first node works as a master node communicating with the client outside. The other nodes work as worker nodes.

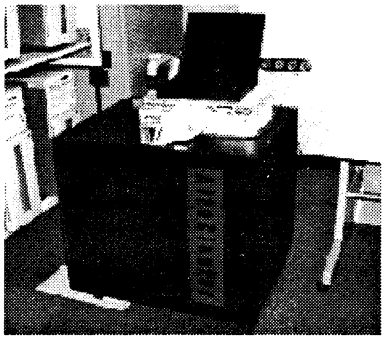
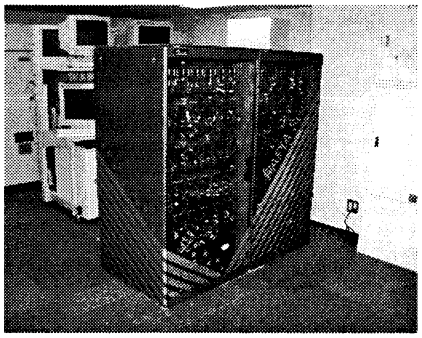



Figure 1: New 8-node PAPIA mini cluster (left), and 64-node PAPIA cluster (right)

	PAPIA mini cluster	PAPIA cluster [1]
No. of Nodes	8 nodes	64 nodes (+2 monitor nodes)
Processor	Dual Pentium II, 450MHz (SpecInt = 16.9)	Pentium Pro, 200MHz (SpecInt = 8.2)
Memory	512MB/node	256MB/node
Disk	6.3GB/node	4.1GB/node
Network	Myrinet 1.28Gbit/sec. + 100-BaseT	Myrinet 1.28Gbit/sec. + 100-BaseT
OS	Linux 2.2.3 + SCore-D	NetBSD 1.2.1 + SCore-D
Physical Size	H635×W770×D635 mm	H1600×W1600×D800 mm
Developed Year	March 1999	Feb. 1998

Table 1: Hardware Specifications of Two PC Clusters

Table 2: PAPIA WWW Service Usage Statistics

	Structure similarity	Sequence homology	Multiple alignment	Secondary structure	TFSEARCH	PDB-REPRDB	WWW Total
Japan	4443	4536	3589	3252	18675	3255	389469
Overseas	322	356	384	491	55978	241	257020
Unknown	526	689	1678	506	29908	69	161422
Total	5291	5581	5651	4249	104561	3565	807911

(Statistics as of November 20, 1999)

Table 3: PAPIA WWW Access Top 20

country	accesses	country	accesses
Japan	389469	Italy	4597
USA	134594	Taiwan	2730
Germany	22389	Israel	2472
UK	16419	Korea	2336
France	12759	Switzerland	2093
Canada	12642	Finland	1973
Netherlands	7452	Norway	1825
Sweden	7436	Belgium	1429
Australia	6737	Austria	1330
Spain	5995	Singapore	1295

(Statistics as of November 20, 1999)

The PAPIA calculation server program resides on each node as a daemon. Any calculation request from the outside will be received as a socket communication (specifying a service and input parameters). The master node dispatches sub-tasks to worker nodes and waits for answers from the workers. In general, dispatching will be repeated until a calculation task is finished. The result list is sorted and edited by the master before it is sent back to the client.

In order to speed-up a database search, each worker node reads the specified (dispatched) part of the database into the local memory at booting. Because of the limited memory capacity, each worker node is in charge of searching only a small fraction of the database. The master node chooses an arrangement for data partitioning. Because each node has an entire copy of the databases on its local disk, the data partitioning and dispatching may be dynamically changed at the cost of loading the data from the local disk. In the current implementation, no pair of nodes works on the same database entry, that is, dispatching is done in a non-redundant way.

Communication between nodes has been implemented through two different methods. The first implementation uses the MPC++ programming language which provides object-oriented remote function invocation. The second implementation uses the MPICH-PM (MPI) communication library for message-passing between processes on different nodes. A drawback of this approach is that a programmer has to design and generate communication packets for transmitting complex protein objects, however, we provide the MPI implementation to guarantee easy porting to other parallel environments.

The parallel efficiency of the PAPIA system is excellent for protein structure similarity searches and protein sequence homology searches, because these parallelized database searches are performed by a simple data partitioning scheme. The PDB protein structure database has more than 10,000, and SWISS-PROT protein sequence database has 80,000 entries. We can assume large granularity in our data partitioning scheme (even if we use a non-redundant representative set of the PDB called the PDB-REPRDB [4] which consists of about 4,000 chains). Overhead for the dispatching and gathering phase is negligible.

Multiple alignment of protein sequences is a difficult problem. All processor nodes must frequently exchange current alignment status. Our method [5] is an extension and parallelization of the Burger-Munson iterative refinement method. It is a kind of parallel hill-climbing search and the best move found by a worker

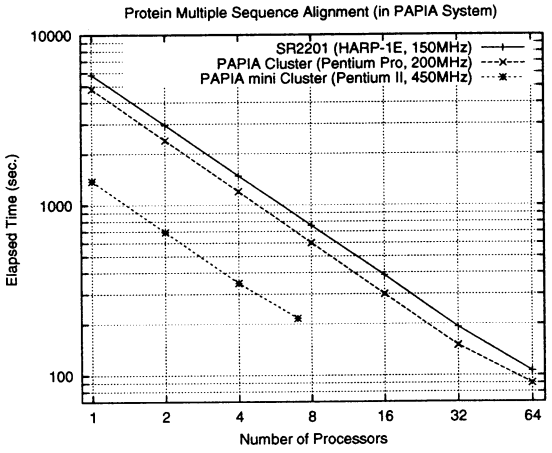


Figure 2: Comparison of Execution Time on the PAPIA mini cluster, the PAPIA cluster, and SR2201: Protein Multiple Sequence Alignment, 64 kinase sequences with 80 amino acids; log-log scale.

(not only the score but also the alignment) must be sent back to the master in each calculation step. Thus the ratio of communication versus calculation is much larger than the simple database scanning cases.

Even for this calculation, experiments showed good parallel efficiency on the PC clusters. Fig. 2 shows the execution time of our precise multiple alignment algorithm for a middle-range example problem (64 *kinase* enzyme sequences, length 80 amino acids). The horizontal axis shows the number of worker nodes (log scale), and the vertical axis shows execution time in seconds (log scale). For the multiple alignment, the PAPIA mini cluster is more than 3-times faster than the old cluster, while the CPU itself (see SpecInt value) is only twice as fast. This comes from the improved bus/memory performance of the Pentium II.

Thus we have succeeded in providing an accurate multiple alignment service, which used to take hours, in an interactive fashion. It will improve the way biologists discover evolutionarily conserved blocks of protein sequences.

We have provided our PAPIA system via a web interface since April 1998. Six services are currently available at http://www.rwcp.or.jp/papia/.

The PAPIA WWW service has been already accessed from 63 different countries. Table 2 shows the usage counts for each calculation service, while 'WWW total' counts include all accesses to the WWW content. Top 20 countries for total WWW accesses are shown in Tab.3. Detailed statistics information is available on the PAPIA WWW page.

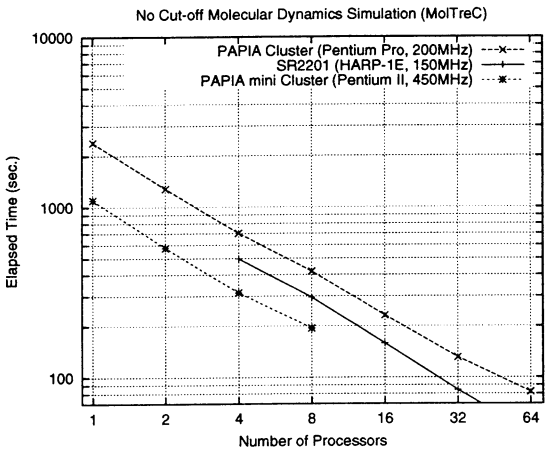


Figure 3: Comparison of Execution Time on the PAPIA mini cluster, the PAPIA cluster, and SR2201: MolTreC No-Cutoff MD Simulation, PROWAT 11585 atoms, time for 100 steps; log-log scale.

2 MolTreC: Parallel No-Cutoff Molecular Dynamics Simulation

We have been also developing original parallel molecular dynamics programs for our PC clusters. **MolTreC** (Molecular Dynamics by Tree Code) [6] is our original parallel molecular dynamics simulator which efficiently performs no-cutoff accurate calculation of Coulomb potentials using a parallelized Barnes-Hut tree code algorithm and the multi time-step method. MolTreC also runs on the PAPIA and now on the PAPIA mini cluster.

Though molecular dynamics of proteins is a communication intensive application, the experimental results showed excellent scalability because of high-bandwidth and low-latency provided by RWCP's PM protocol on Myrinet. Fig. 3 shows the execution time for 100 simulation steps of MolTreC on the well-known PROWAT (plastocyanin in water) benchmark with 11585 atoms. The horizontal axis shows the number of worker processors (log scale), and the vertical axis shows execution time in seconds (log scale). In the MolTreC simulation, the performance gap between two clusters is smaller than the case of Fig. 2 because of a fixed cost for heavy communication load. SR2201 showed relatively better performance because the MolTreC program is written in FORTRAN77 and is suitable for the pseudo-vector functionality of SR2201. PROWAT Benchmark results for other machines can be found on the AMBER WWW page at http://www.amber.ucsf.edu/amber/.

3 Conclusion

The PAPIA system running on our PC cluster enables very fast computational analysis of proteins. A calculation which used to take hours is now carried out in minutes. PAPIA WWW service has been used by biologists world-wide. The wide memory space provided by the cluster allows on-memory database searches, and local disks provide flexibility in parallel database searches.

Molecular dynamics simulation of proteins is a communication intensive application. However, our PC cluster showed excellent scalability. With the SCore multi-user parallel environment developed by RWCP [7], we can even run the PAPIA system and MolTreC (or any other application) simultaneously on a cluster.

The PC cluster approach is an easy and quick way to fully utilize the power of the latest microprocessors. PC clusters will become a powerful but inexpensive tool for biochemical and other scientific researches.

References

[1] Akiyama, Y., Onizuka, K., Noguchi, T., *et al.*: Parallel Protein Information Analysis (PAPIA) system running on a 64-node PC Cluster, *Proc. 9th Genome Informatics Workshop*, pp.131–140 (1998).

[2] Akiyama, Y: Protein Information Analysis on Parallel Computers, *Journal of Japanese Society for Artificial Intelligence*, Vol. 15, No. 1 (2000). [in Japanese]

[3] Onizuka, K., Noguchi, T., and Akiyama, Y.: Parallel PDB Data Retriever 'PDB Diving Booster', *Lecture Notes in Computer Science*, Vol. 1336, pp. 389–396, Springer-Verlag (1997).

[4] Noguchi, T., Onizuka, K., Akiyama, Y., and Saito, M.: PDB-REPRDB: A Database of Representative Protein Chains in PDB (Protein Data Bank), *Proc. 5th Intelligent Systems for Mol. Biol.*, pp. 214–217 (1997).

[5] Totoki, Y., Akiyama, Y., Onizuka, K., *et al.*: Multiple Protein Sequence Alignment using Parallel Iterative Algorithm and A* Algorithm, *Trans. of IPSJ*, Vol. 40, No. SIG9 (TOM2) (1999). [in Japanese]

[6] Misoo, K., Shizawa, Y., Akiyama, Y., *et al.*: Development of Molecular Dynamics Programs for Proteins with Parallelized Barnes-Hut Tree Code, *Proc. 1999 Joint Sympo. on Parallel Processing (JSPP'99)*, pp. 143–150 (1999). [in Japanese]

[7] Ishikawa, Y., Tezuka, H., Hori, A., *et al.*: RWC PC Cluster II and SCore Cluster System Software - High Performance Linux Cluster, *Proc. 5th Annual Linux Expo*, pp. 55–62 (1999), (see also http://pdswww.rwcp.or.jp/).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Inference of Nonlinear Biological Systems by Using Linear Programming

Tatsuya Akutsu* Satoru Miyano* Satoru Kuhara#

*Human Genome Center, Institute of Medical Science, University of Tokyo
4-6-1 Shirkanedai, Minato-ku, Tokyo 108-8639, Japan
{takutsu,miyano}@ims.u-tokyo.ac.jp

Graduate School of Genetic Resources Technology, Kyushu University
6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan
kuhara@grt.kyushu-u.ac.jp

Abstract

Inference of genetic networks and metabolic networks from time series data obtained by biological experiments using DNA micro arrays is an important topic in bioinformatics. In this article, we propose a method for inferring S-systems from time series data, where S-systems are based on a particular kind of nonlinear differential equation and have been applied to the analysis of various biological systems. We reduce each instance of the inference problem to the linear program by means of focusing on the sign of differential (or finite difference). Although the errors of estimated parameters are relatively large, the proposed method is much faster than most other methods that are based on nonlinear optimization.

keywords: genetic network, DNA microarray, linear programming, S-system

1 Introduction

Due to the recent progress of the *DNA microarray* technology [1], it has become possible (to some extent) to measure the gene expression levels of most of the genes of an organism simultaneously. Recently, many studies have been done in order to develop computational methods for reconstructing underlying *genetic networks* from time series data of gene expression patterns.

Several studies have been done using the Boolean network [2], where a gene takes one of two states (ON or OFF), and a gene regulation rule is given as a Boolean function. Liang *et al.* [2] developed the REVEAL algorithm (reverse engineering algorithm) for inferring genetic networks from state transition tables which correspond to time series data of gene expression patterns. We also made several theoretical studies on inference of Boolean networks [3]. Since genetic networks are very complex, there are many criticisms on the Boolean network approach. Of course, many other models and inference methods have been proposed: a qualitative model [4], hybrid models [5, 6], a statistical method for inference of infer chemical networks [7], and models and inference methods based in linear differential equations [8, 9]. However, no method seems to be sufficient.

On the other hand, the S-system (*synergistic* and *saturable* system) has been developed for analyzing and modeling biological systems [10], where S-systems are based on a particular kind of nonlinear differential equation. S-systems have been successfully applied to the analysis of various biological systems [10]. Recently, Tominaga and Okamoto [11] applied GA (Genetic Algorithm) to inference of S-systems. However, their method was time consuming and was limited to inference of S-systems with a few parameters.

In this article, we propose a novel method for inferring S-systems from time series data. We reduce each instance of the inference problem to a linear program by means of focusing on the sign of differential (or finite difference). We call this method the LP-based method, where LP denotes *linear programming* in this article. Although the errors of estimated parameters are relatively large, the LP-based method is much faster than the GA-based method. It is also expected that the LP-based method is much faster than most other methods that are based on nonlinear optimization because linear programs can be solved in polynomial time.

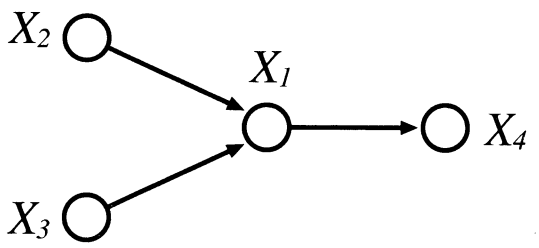


Figure 1: Genetic network corresponding to an S-system defined by $\frac{dX_1}{dt} = \alpha \cdot (X_2^2 + X_3^3) - \beta \cdot X_4^{2.5}$.

2 Inference of S-systems

Here we briefly review the definition of the S-system [10, 11]. Let $\{X_1, \ldots, X_n\}$ be a set of genes and/or chemical substances in the underlying biological network. Let $X_i(t)$ be the value (expression level or concentration) of a gene or a chemical substance X_i at time t.

An S-system is a set of *nonlinear* differential equations of the form

$$\frac{dX_i(t)}{dt} = \alpha_i \prod_{j=1}^{n} X_j(t)^{g_{i,j}} - \beta_i \prod_{j=1}^{n} X_j(t)^{h_{i,j}}$$

where α_i and β_i are multiplicative parameters called *rate constants* and $g_{i,j}$ and $h_{i,j}$ are exponential parameters called *kinetic orders*.

The inference problem is, given time series data $X_i(t)$ that are assumed to be generated from an S-system $\mathcal{S}$, to estimate to parameters α_i, β_i, $g_{i,j}$ and $h_{i,j}$ of $\mathcal{S}$.

Since S-systems are nonlinear, we can not apply linear regression [9] to inference of S-systems. As mentioned in the introduction, Tominaga and Okamoto [11] applied GA (Genetic Algorithm) to inference of S-systems with a few parameters. However, their method was time consuming. Therefore, we developed a new method for the inference of S-systems based on LP.

The method is quite simple. Assume that $\frac{dX_i(t)}{dt} > 0$ at time t. By taking 'log' of each side of $\alpha_i \prod X_j(t)^{g_{i,j}} > \beta_i \prod X_j(t)^{h_{i,j}}$, we have

$$\log \alpha_i + \sum_{j=1}^{n} g_{i,j} \log X_j(t) > \log \beta_i + \sum_{j=1}^{n} h_{i,j} \log X_j(t) .$$

Since $X_j(t)$'s are known data, this inequality is linear if we treat $\log \alpha_i$'s and $\log \beta_i$'s as parameters. In the case of $\frac{dX_i(t)}{dt} < 0$, we can obtain a similar inequality. Therefore, solving these linear inequalities by LP, we can determine parameters.

However, parameters are not determined uniquely even if a lot of data are given, because the inequality can be re-written as $(\log \alpha_i - \log \beta_i) + \sum (g_{i,j} - h_{i,j}) \log X_j(t) > 0$. Therefore, only relative ratios of $\log \alpha_i - \log \beta_i$ and $g_{i,j} - h_{i,j}$'s are determined (for each i). But, this information is useful for qualitative understanding of S-systems. Since $\prod X_j(t)^{g_{i,j}}$ contributes to the net production of X_i, $\prod X_j(t)^{h_{i,j}}$ contributes to the net degradation of X_i and it is not usual that X_j contributes to both the net production and the net degradation, either $g_{i,j} = 0$ or $h_{i,j} = 0$ holds for each (i, j) in most cases. Thus, the fact that $|g_{i,j} - h_{i,j}|$ is large means that X_i is influenced by X_j.

It should be noted that the LP-based method is not robust for noises since parameter values are not determined even if one linear inequality is not satisfied. However, such noisy cases may be handled by using *robust linear programming* [12].

3 Computational Experiments

We made computational experiments on the LP-based method using a SUN ULTRA-2 Workstation with 1 CPU (296MHz). In order to solve LP, we used SOPT [13]. In these experiments, we used $\frac{\Delta X(t)}{\Delta t}$ in place of $\frac{dX(t)}{dt}$.

First we examined the following simple cases of $n = 2$, where case (A) was examined in Ref. [11] too.

	i	α_i	$g_{i,1}$	$g_{i,2}$	β_i	$h_{i,1}$	$h_{i,2}$
(A)	1	3.0	0.0	-2.5	3.0	0.125	0.0
	2	3.0	2.5	0.0	3.0	0.0	0.125
(B)	1	3.0	0.0	-2.5	3.0	1.25	0.0
	2	3.0	2.5	0.0	3.0	0.0	1.25

As input data, time series data beginning from randomly generated initial values in $[0.5, 2.0]$ were used. The Euler method was used to generate the time series data, where $\Delta t = 0.02$ was used. Since the LP-based method can only compute relative values of $g_{i,j} - h_{i,j}$'s, we compare the ratios $r_1 = \frac{g_{1,1} - h_{1,1}}{g_{1,2} - h_{1,2}}$ and $r_2 = \frac{g_{2,2} - h_{2,2}}{g_{2,1} - h_{2,1}}$. The following table shows the result, where average values and standard deviations over 20 trials are shown. m denotes the total number of time points in the data, where 50 point data are generated from each set of initial values.

Table 1: Simulation result on inference of S-systems with two genes

		Correct	$m = 1 \times 50$	$m = 5 \times 50$	$m = 10 \times 50$
(A)	(r_1, σ)	(0.05, -)	(0.129, 0.032)	(0.081, 0.009)	(0.077, 0.011)
	(r_2, σ)	(-0.05,-)	(-0.261,0.232)	(-0.086,0.023)	(-0.085,0.011)
(B)	(r_1, σ)	(0.5, -)	(0.653, 0.099)	(0.598, 0.054)	(0.574, 0.040)
	(r_2, σ)	(-0.5,-)	(-0.648,0.108)	(-0.568,0.032)	(-0.538,0.029)

Table 2: Simulation result on inference of S-systems with 10 nodes

	$m = 25 \times 20$	$m = 50 \times 20$	$m = 100 \times 20$
$K = 2$	30%	86%	100%
$K = 4$	26%	69%	87%

In each case, parameters were inferred within 1 second, which is much faster than the GA-based algorithm [11]. On the other hand, the errors (in case (A)) are larger. But, it is not a serious problem because we do not aim at determining precise values. We only want to know whether each $|g_{i,j} - h_{i,j}|$ is relatively large or small. Note that the errors are small for $m = 50$ in case (B), whereas the errors are not small even for $m = 500$ in case (A). This observation suggests that good values are not inferred if parameters in the different levels are included.

Next we examined whether or not qualitative relations are correctly inferred, by applying the LP-based method to the case of $n = 10$ and $K = 2$ and the case of $n = 10$ and $K = 4$. Note that only the case of $n = 2$ was examined in Ref. [11]. In these cases, we did not try to infer precise values of parameters, but we tried to infer whether or not X_i is influenced by X_j examining the value of $|g_{i,j} - h_{i,j}|$. We say that the set of input nodes $\{X_{i_1}, \cdots, X_{i_K}\}$ to X_i is *correctly inferred* if the LP-based method outputs the same set for X_i, where we say that X_j is an input node to X_i if $h_{i,j} \neq 0$ and $g_{i,j} \neq 0$ hold in the original S-system. We count the number of nodes for which the sets of input nodes are correctly inferred. The result is shown in the table below. In the table, the average ratios (%) of correctly inferred nodes over 10 randomly generated S-systems are shown, where the following values are used: $\Delta t = 0.01$, $\alpha_i = \beta_i = 3.0$, $0.5 < |g_{i,j}| < 3.0$, $0.5 < |h_{i,j}| < 3.0$. Even in the case of $m = 100 \times 20$, each inference can be done within 30 sec. (CPU time). From this table, it is seen that the sets of input nodes are correctly inferred for most nodes if m is large enough.

Finally, we examined the case of $n = 100$, $K = 4$, and $m = 1000 \times 20$. In this case, the LP-based method inferred the sets of input nodes correctly for 96 nodes using less than 5 hours (with 1 CPU), where $\Delta t = 0.005$. This result demonstrates the power of the LP-based method because we are tackling a very hard problem, inference of nonlinear systems with more than $100 \times 100 \times 2$ parameters.

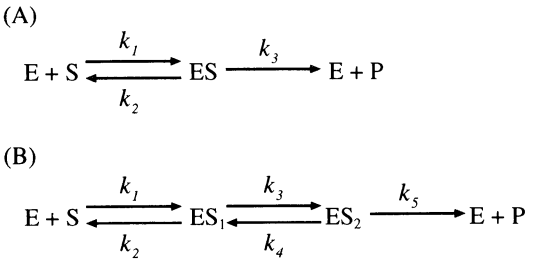


Figure 2: Reaction (A) can be expressed as an S-system, whereas reaction (B) can not be expressed as an S-system.

4 Conclusions

In this paper, we proposed an LP-based method for inferring an S-system from time series data. This LP-based method is much faster than the GA-based: it took less than one second in order to compute parameters for an S-system with two nodes, whereas the GA-based method took about two hours.

However, it requires many time series data beginning from different sets of initial values, where different

sets correspond to different environments or different conditions. Since time series data of 7 or 17 points beginning from a few different sets of initial values were only available [1], we could not apply the proposed methods to real data. It seems almost impossible to get more information than those obtained by clustering, if only a small number of time series data can be used. However, many biological experiments are currently being done using gene disruptions and gene overexpressions, and it is expected that a large number of more precise data will be available in the near future. For example, several hundreds of disruptants of *Saccharomyces cerevisiae* are being made by the group to which the third author of this paper belongs. Therefore, the assumption of existence of times series data beginning from many initial value sets will become realistic in the near future. Of course, much faster algorithms should be developed for handling a large amount of data.

Another drawback of the proposed methods is that complex enzymatic reactions (for example, three-stage enzymatic reactions) can not be handled: these reactions can not be represented in the form of the S-system. Therefore, development of the methods to infer complex enzymatic reactions is important future work.

Acknowledgements

This work was partially supported by the Grant-in-Aid for Scientific Research on Priority Areas, "Genome Science", of the Ministry of Education, Science, Sports and Culture in Japan.

References

[1] DeRisi JL, Lyer VR, Brown PO (1997), Exploring the metabolic and genetic control of gene expression on a genomic scale. *Science* 278:680-686

[2] Liang S, Fuhrman S, Somogyi R (1998), REVEAL, a general reverse engineering algorithm for inference of genetic network architectures. *Proc. Pacific Symp. on Biocomputing* 3:18-29

[3] Akutsu T, Miyano S, Kuhara, S (1999), Identification of genetic networks from a small number of gene expression patterns under the boolean network model. *Proc. Pacific Symp. on Biocomputing* 4:17-28

[4] Thieffry D, Thomas R (1998), Qualitative analysis of gene networks. *Proc. Pacific Symp. on Biocomputing* 3:77-88

[5] McAdams HH, Shapiro L (1995), Circuit simulation of genetic networks. *Science* 269:650-656

[6] Yuh C-H, Bolouri H, Davidson EH (1998), Genomic Cis-regulatory logic: experimental and computational analysis of a sea urchin gene. *Science* 279:1896-1902

[7] Arkin A, Shen P, Ross J (1997), A test case of correlation metric construction of a reaction pathway from measurements. *Science* 277:1275-1279

[8] Chen T, He HL, Church GM (1999), Modeling gene expression with differential equations. *Proc. Pacific Symp. Biocomputing* 4:29-40

[9] D'haeseleer P, Wen X, Fuhrman S, Somogyi R (1999), Linear modeling of mRNA expression levels during CNS development and injury. *Proc. Pacific Symp. Biocomputing* 4:41-52

[10] Irvine DH, Savageau MA (1990), Efficient solution of nonlinear ordinary differential equations expressed in S-system canonical form. *SIAM J. Numer. Anal.* 27:704-735

[11] Tominaga D, Okamoto M (1998), Design of canonical model describing complex nonlinear dynamics. *Proc. IFAC Int. Conf.* CAB7:85-90

[12] Bennett KP, Mangasarian OL (1992), Robust linear programming discrimination of two linear separable sets. *Optimization Method and Software* 1:23-34

[13] SAITECH Inc. (1998), *Smart Optimizer User's Guide*

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Bio-computing in the 21st Century

Masanori Arita
Electrotechnical Laboratory
Umezono 1-1-4, Tsukuba-shi
305-8568 Ibaraki, Japan

Abstract

Bio-computing has a potential to become a mainstream science in a near future. In this interdisciplinary field, a new paradigm is awaited to clarify the mechanism of protein interaction and, eventually, the mystery of life. This short paper is intended as an introductory guide to the ongoing two projects: simulation of metabolism and molecular computation. In both areas, collaboration between natural science and engineering is expected.

1 Introduction

'Genome' is a man-made word from gene and chromosome, and means genetic information within a cell as a whole. Within a few years, the milestone of this genetic information will be soon achieved: the database of 3 billion bases of human DNA will become public. In fact, more than 20 species of bacteria have been sequenced and the information is already available.

Genome informatics started as a field to study this catalogue of genes, and has been providing various tools for the genome analysis [1]. The whole genome analysis drastically changed the living around us: genetic engineering of a bacterium or plant now references its metabolism, and the modification of genes is no longer done in hit-or-miss manner. Its too rapid development has recently induced an antipathy against genetically engineered foods, but the advantage and benefit of such products is inarguable. The engineering of disease gene is also on the way. For example, it is known that some genetic diseases can be treated 'molecularly' using artificial viruses. The new flu vaccine is also one of such victories. Thus, we are witnessing many achievements even before the full-scale analysis of human genes, on the engineering as well as on the commercial side. Clearly, the direction of information flow is from the analytic study in molecular biology to the new products in genetic engineering. What is expected now is the reverse flow, i.e. engineering to biology.

One of the original aims of genome informatics has not been fulfilled yet: elucidation of life. Although researchers are aware that sequence information only is not sufficient for this purpose, the complexity and ambiguity of protein interaction makes its analysis prohibitively hard. Our weakness resides in the lack of simple and beautiful model in protein interaction, comparable to the hybridization or the encoding system of DNAs. We can describe many properties of DNAs with four letters (A,C,G,T), but we cannot express protein functions in a similar way. What we need are new models to describe proteins, and eventually, to describe cellular functions including life.

Even with today's advanced technology, information obtained from laboratory experiments is often ambiguous and erroneous. This is why experimental biologists have not been convinced to do combinatorially many, comprehensive experiments instead of intuitive, trend-driven ones: experimental results are intrinsically qualitative, and the simple-minded compilation of data is not enough to logically deduce a meaningful model. Mainly for this reason, genome informaticians could not have beaten models proposed by biologists, and no model has been deduced or computed by a computer.

The stage is set. The traditional approach is facing a difficulty in analyzing cellular mechanism. Analytic study is not sufficient to output models from qualitative data. In order to cope with this challenge, more engineering-oriented perspective is necessary. A new paradigm should set a broader, more comprehensive field, called **bio-computing**. Without such paradigms, protein analysis (and other advanced analyses of life) would be a daunting effort with no prospect. This paper introduces two ongoing projects toward bio-computing.

2 Metabolic Simulation

Metabolism is a network of chemical reactions catalyzed by enzymes [2]. For example, the sugar and protein in our food are digested into water and carbon dioxide, and the obtained energy is used to produce adenosine triphosphate (ATP). On the other hand, building blocks of a cellular structure are synthesized from smaller compounds using ATP. This recurrence of synthesis and degradation with chemical energy is one important requirement of life, and its analysis is academically important too. In this work, we propose a modeling of metabolism with chemical reactions only, i.e. without information of enzymes themselves.

The strategy is to use a graph algorithm to reconstruct metabolic pathways: the entire metabolism is represented as a graph, in which compounds and enzymatic reactions correspond to nodes and edges, respectively [3, 4]. For this representation, each chemical reaction is interpreted as in Figure 1. Note that each reaction corresponds to multiple edges connecting atoms of compounds. By looking at this graph of metabolism, we can find out how a certain compound (e.g. a natural medicine) is synthesized and degraded, or how to enhance the metabolism by genetic engineering for a particular purpose (e.g. for synthesizing medicines).

Finding metabolic pathways corresponds to selecting subgraphs out of the large metabolic graph by some restrictions, such as being a linear path or a cycle. The subgraphs to be extracted can be computed by algorithms such as k-shortest paths [5]. Important point is that the program should output a collection of possible candidate paths, not a single optimal path.

2.1 For Interested Readers

The authoritative model of metabolism is called 'metabolic map', in which hundreds of known metabolism connecting chemical molecules is drawn like railway maps. However, genomic sequences of bacteria pose several questions on this metabolic map. For example, after assigning the putative gene functions computed from sequence similarity in Eschelichia coli (colon bacillus), important enzymes in the pathway to synthesize amino acid lysine are missing. The answers for this kind of question are necessary in understanding the mechanism of life.

Metabolic information can be obtained from Kyoto Encyclopedia of Genome and Genes (KEGG) database [6]. Several other on-line databases also provide information on enzymes and the metabolism of bacteria.

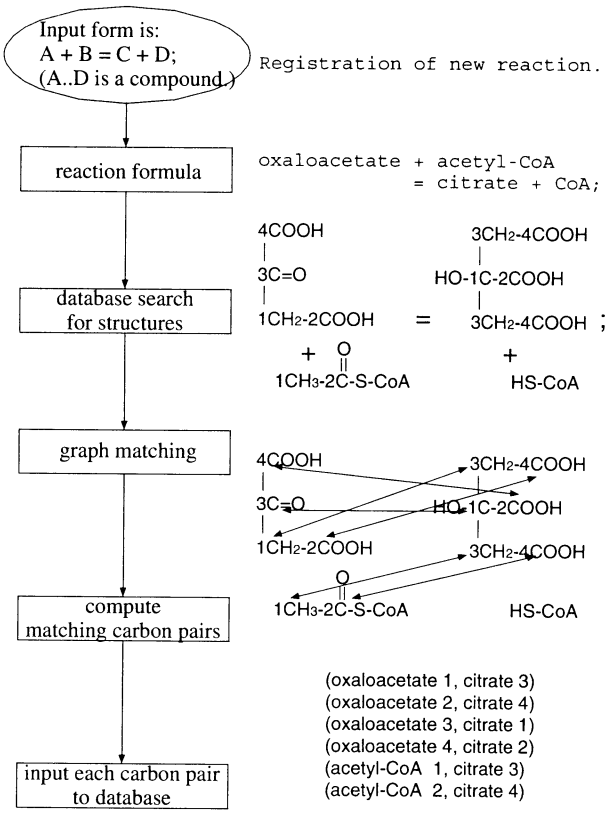


Figure 1: Flowchart for modeling chemical reactions.

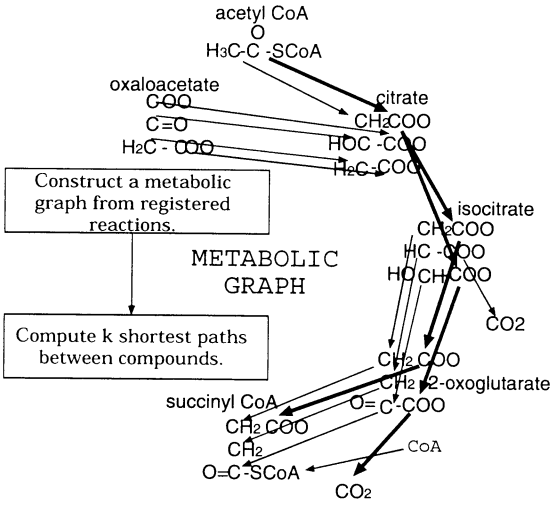


Figure 2: Tracing result of a carbon in acetyl CoA in TCA cycle. Bold-arrows are the mapping from the single carbon labeled in acetyl CoA. Such paths can be extracted using graph algorithms.

3 Molecular Computing

Molecular computing is a radical paradigm of computation, in which chemical molecules are used to solve computational problems. It is the reverse approach of the traditional genome informatics, in which computers are used to analyze molecules. Molecular computing has its root on DNA computing, which was sensationally introduced in 1994 by L. Adleman [7]. He demonstrated that DNAs can be used to solve Hamiltonian Path Problem.

DNA is normally regarded as an information carrier in biology, but it is an information processor too, if we consider its role within a cell. After Adleman's work, researchers have investigated the computational power of DNAs, and many algorithms and experiments have been reported [8].

Molecular computing is a natural extension of DNA computing. It aims the elucidation of computational power of living molecules in general, and also aims the development of information processing units with molecules. The field is an extension three existing areas: computer science, mechanical engineering, and molecular biology. Their relationship can be described as a tetrahedron in Figure 3.

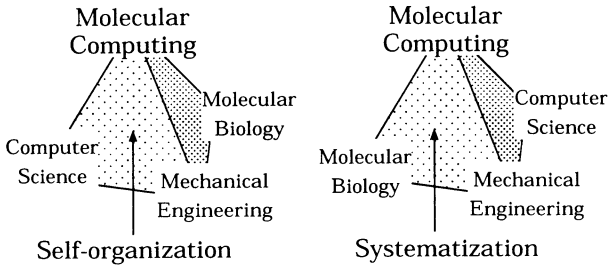


Figure 3: Relation between three established areas. Three faces of the tetrahedron correspond to: 1. self-organization (from computer science and mechanical engineering), 2. systematization (from mechanical engineering and molecular biology), and 3. molecular evolution (from molecular biology and computer science).

As an example of DNA computers, we briefly introduce the 'whiplash PCR' mechanism. PCR stands for polymerase chain reaction, and is the standard laboratory procedure to amplify DNA sequences. We found that PCR process can be used to simulate a state-machine implemented in DNAs [9].

In Whiplash PCR, each DNA strand performs like a small Turing machine. As in Figure 4, each DNA

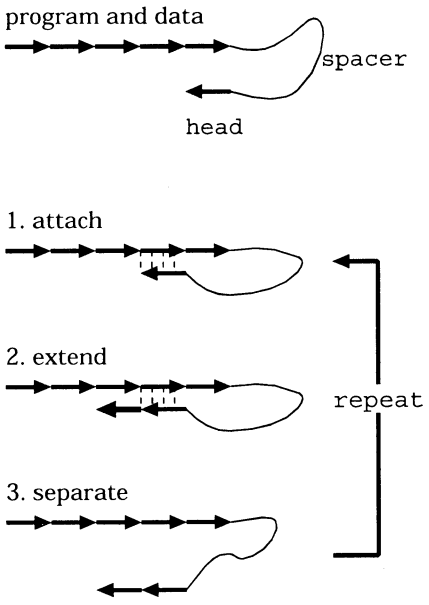


Figure 4: The evaluation steps in whiplash PCR.

consists of three parts: program, data, and head. Like a Turing machine, the head accesses a particular location in the program (or data), copies one symbol, and is detached. In the next cycle, the copied symbol becomes the head, and it will access another location. The flapping movement is regulated by the reaction temperature, and the termination after copying is skillfully implemented in the DNA sequence. Since we can design and synthesize arbitrary DNA sequences, we can encode different programs to each DNA. This opens up a possibility of parallel evaluation of very large number of DNA machines. This mechanism has been shown to solve vertex-cover problem and satisfiability problem, although in a small scale [10].

3.1 For Interested Readers

Molecular computing considers molecules to be parts of a life system. Although we are not the first to propose this paradigm, a reliability test or a performance analysis of molecular interaction has not been done in a systematic way. Recent popularity of DNA chips (or DNA arrays) shed light on the use of molecules as an analyzing tool, and biologists are now interested in the possible operations on DNAs [11].

The analysis of life 'as a system' is also started in biological simulation projects. Several projects try to simulate a living cell using the data from laboratory

experiments. The major ones in Japan are E-CELL project at Keio University and Kitano Symbiotic Systems Project at ERATO [12, 13].

4 Future Bio-Computing

The above two approaches are not exclusive in that biology is systematically analyzed, and would constitute fundamental technology in bio-computing. The approach will inevitably model-based, because experimental data are not enough to logically deduce new findings. It is not important whether the template models are made by biologists or by computer scientists. The crucial point is we require various (even non-realistic) models, so as not to be stuck in a narrower vision. The main driving force of bio-computing has been computer scientists, but more biologists and engineers are expected join to clarify and utilize the biological systems. Readers are encouraged to follow links in the bibliography.

Acknowledgments

The work is supported by Grant-in-Aid for Scientific Research on Priority Area "Genome Informatics," from Ministry of Education, Science, Sports and Culture, Japan. This work is also supported by the Japan Society for the Promotion of Science "Research for the Future" Program (JSPS-RFTF 96I00101).

References

[1] GenomeNet Server `http://www.genome.ad.jp`
Major databases and analyzing tools are collected here. Introductory guide to genome informatics is at `http://www.genome.ad.jp/Japanese`. Several books on genome informatics are published, but are not recommended.

[2] J. G. Salway, *Metabolism at a Glance*, Blackwell science, 1994.
This is the best book for a beginner on metabolism. For a comprehensive map, *Biochemical Pathways* edited by G. Michal (Wiley, Spektrum 1999) is a published version of the famous poster on metabolism from Böhringer Mannheim pharmaceutical company, which is now merged with Roche.

[3] M. Arita, "Automated Metabolic Reconstruction", Ph.D. Thesis, Tokyo University, 1999.

[4] M. Arita, K. Asai, T. Nishioka "Finding Precursor Compounds in Secondary Metabolism", *Proc 10th Genome Informatics*, 113–120, 1999.

[5] D. Eppstein, "Finding the k Shortest Paths", *Proc 25th Ann Symp on Foundations of Comp Sci (FOCS'94)*, 154–165, 1994.

[6] H. Ogata, S. Goto, *et al.*, "KEGG: Kyoto Encyclopedia of Genes and Genomes", *Nucleic Acids Res* **27**(1):29–34, 1999.
(available at `http://www.genome.ad.jp/kegg`)
KEGG is a digitized version of the popular metabolic map, and provides the best interface among on-line databases such as UMBBD (`http://www.labmed.umn.edu/umbbd/index.html`) or WIT (`http://wit.mcs.anl.gov/WIT2`).

[7] L. Adleman, "Molecular Computation of Solutions to Combinatorial Problems", *Science* **266**:1021–1024, 1994.
For more recent progress, see report data of DARPA project at `http://bmc.cs.duke.edu/`.

[8] M. Hagiya and A. Nishikawa, *DNA Computing* (DNA keisan), A review to appear in Genetic Algorithms Vol.4, Sangyo Tosho (in Japanese).
For other reviews and related papers of the project in Japan, see the bibliography at `http://nicosia.is.s.u-tokyo.ac.jp/moco`.

[9] M. Hagiya, M. Arita, *et al.*, "Towards Parallel Evaluation and Learning of Boolean μ-Formulas with Molecules", *DIMACS Series in Discr Math and Theor Comp Sci* Vol.**48**:57–72, 1999.

[10] K. Sakamoto, D. Kiga, *et al.*, "State Transitions by Molecules", *Proc 4th Meeting on DNA Based Computers*, 1998.

[11] A supplement of *Nature Genetics* **21**(1), 1999.
This special issue features the advances in DNA chips.

[12] M. Tomita, K. Hashimoto, *et al.*, "E-CELL: software environment for whole-cell simulation", *Bioinformatics* **15**(1):72–84, 1999.

[13] Kitano Symbiotic Systems Project focuses on the simulation of four species: fruitfly, nematode, yeast, and colon bacillus. Started in late 1998, the project is looking for ambitious young researchers. For positions, visit `http://www.symbio.jst.go.jp/~yoko`.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Analysis of Viral Evolution Pattern by Using Mixed Evolution Model

Fengrong Ren, Soichi Ogishima and Hiroshi Tanaka
Medical Research Institute, Tokyo Medical and Dental University
1-5-45 Yushima, Bunkyo-ku, Tokyo 113-8510, Japan
Email: {rencom@mri.tmd.ac.jp, ogishima@gentzen.mri.tmd,ac.jp, tanaka@cim.tmd.ac.jp }

Abstract

A mixed evolution model for the evolution of viral sequences is presented for use in phylogenetic analysis. A codon-based model, which used a Markov process to describe base substitutions between codons, is employed to calculate nonsynonymous and synonymous substitution numbers. By incorporating this codon-based model, positive selection and neutral evolution that are supposed to exist in the process of viral evolution can be distinguished and estimated. Furthermore, a star-linking approach is developed in this study and a longitudinal phylogenetic tree can be reconstructed from sequential molecular data. Our model is applied to a data set of the V3 region of the HIV-1 envelope gene that is sequenced at different years after the infection of one single patient. The results suggest that mixed evolution model may provide more realistic model than traditional evolution models for the analysis of viral evolution, because it can estimate both neutral and positively selective evolution patterns and reconstruct a longitudinal phylogenetic tree from sequential molecular data.

Keywords: viral evolution, positively selected evolution, neutral evolution, mixed evolution model, star-linking approach.

1 Introduction

Ever since the neutral theory was proposed by Kimura in 1968 [1], the mechanism of evolution in molecular level has been thought to be mostly neutral to the natural selection, which is different from Darwin's theory of evolution that is valid in morphological level. The neutral theory asserts that most of molecular variation is due to random fixation, where random drift rather than natural selection dominates their evolution. However, the facts that positive selection was also observed in molecular level have been reported by many biologists recently, so that this issue are hotly debated and studied and a number of methods have been developed [2] [3] [4] [5] [6]. But, these papers only studied how to detect and estimate the positively selective evolution that occurs on molecular sequences, and not studied that how the neutral evolution and positively selective evolution behave as time goes on, namely how these two evolution patterns behave in evolution process. Therefore, these methods are not sufficient for analysis of molecular evolution and can not to be applied to reconstruct molecular phylogenetic tree.

In this study, we develop a new model for analysis of viral evolution. We call this model as "mixed evolution model", because it can estimate both neutral and positively selected evolution. It is well known that the evolutionary rate of virus is very rapid compared to most DNA genomes, especially the evolutionary rate of retrovirus is valued to be millionfold greater than that of host DNA genomes. This feature of virus is quite appropriate for analysis of molecular evolution. In fact, the studies mentioned above almost used molecular data of virus. To assess the efficiency of our method developed in this paper, we applied it to a data set of the V3 region of HIV-1 envelop gene and carried out a detailed analysis of the process of viral evolution.

2 Data and Method

2.1 Data

The sequences to be analyzed in this study are from GenBank, which were sequentially isolated over 7 years after HIV-1 infection of one single patient [7]. In Holmes's original paper, there were a total of 24 different V3 loop sequences to be analyzed and they were assigned a letter (A-F). In year 0, only one sequence was observed and assigned a letter A. No data were available for years 1 and 2. The other 23 sequences were obtained from year 3 to year 7 (B, C1-5,

D1-8, E1-8 and F). But only 15 different nucleotide sequences were available from GenBank. They are sequence A for year 0, sequences C1, C2, C3, C4, C5 and E1 for year 3, sequences D2 and E2 for year 4, sequences E2, E4 and E5 for year 5, sequences D6, D7, D8 and E7 for year 6, and sequences D7 and D8 for year 7. In this study, we only analyze these 15 viral sequences obtained at different time points. .

2.2 Method

2.2.1 Mixed Evolution Model

As the all of present models for analysis of molecular evolution are developed according to neutral evolution theory, the positive selection can not be distinguished and estimated by them. Our mixed evolution model can resolve this problem. In general, the excess of nonsynonymous substitution over synonymous substitution numbers is an indicator of positively selective evolution, so that the estimation of nonsynonymous substitution rate (d_n) and synonymous substitution rate (d_s) is important to distinguish these two kinds of evolution pattern. For calculating this ratio, Yang's codon-based model [8] is employed in this study. The codon-base model is obtained by modeling at the codon level instead of the nucleotide or amino acid levels and thus it is more realistic model compared to other evolution models. According to this model, stop codons are not allowed in the gene sequence and substitutions between sense codons are described by a continuous-time Markov process. The instantaneous substitution rate from codon i to j ($i \neq j$) is given by

$$Q_{ij} = \begin{cases} 0, & \textit{if the two codons differ at more than one position,} \\ \pi_j, & \textit{for synonymous transversion,} \\ \kappa\pi_j, & \textit{for synonymous transition,} \\ \omega\pi_j, & \textit{for nonsynonymous transversion,} \\ \omega\kappa\pi_j, & \textit{for nonsynonymous transition.} \end{cases}$$

Where parameter κ is the transition /transversion rate ratio, ω is the nonsynonymous/ synonymous rate ratio ($\omega = d_n/d_s$), and π_j is the equilibrium frequency of codon j calculated using the nucleotide frequencies at the three codons positions. If $\omega \leq 1$.the viral sequences are thought to be having the neutral evolution, to the contrary, if $\omega > 1$ then the viral sequences are thought to be having positive selected evolution.

2.2.2 Star-linking Approach

For estimating nonsynonymous and synonymous substitution rates, we employed Yang's codon-based model. But, since this model is made only for calculating nonsynonymous and synonymous substitution numbers, it can not construct an evolutionary phylogenetic tree from molecular data. Therefore, we developed a star-linking method to reconstruct the phylogenetic tree and applied it to the above 15 sequences of HIV-1. First, we assume a star-shape topology for estimating the relationship of sequences including quasi-species that are obtained at time point n and time point $n+1$ (see Fig.1), and then a resultant topology will be constructed. In this resultant topology, if some viral sequence is located at node m, this sequence obviously has the evolutionary relationship with other viral sequences. Next, based on that topology obtained, a further combined topology is assumed to estimate the relationship of the sequences that are obtained from next time point $n+2$, and then a further combined resultant tree can be constructed. The relationship of sequences obtained from other time points ($n+2, n+3, n+4, n+5 \ldots$) are likewise estimated and finally the whole phylogenetic tree is reconstructed by linking each resultant topology. The evolutionary relationship of 15 viral sequences under the analysis will be shown in this reconstructed tree.

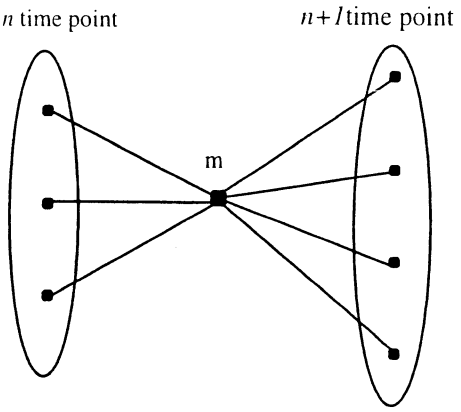


Figure 1: Star-linking model used in this study

3 Results

The whole resultant tree reconstructed by our mixed evolution model is shown in Fig.1. The tree topology is nearly consistent with Holmes's that obtained by calculating the differences between the amino acid sequences. Since we incorporate codon-based model in our method, not only the tree topology is constructed, but the evolution patterns in different

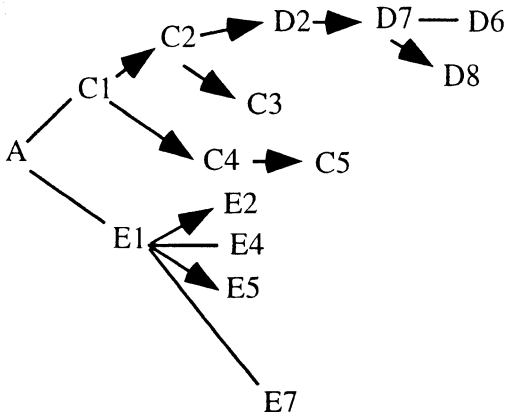


Figure 2: Longitudinal phylogenetic tree reconstructed by mixed evolution model from 15 molecular sequences of HIV-1.Allowheads represent the positive selection

Table 1: Nonsynonymous substitution rates (d_n) and synonymous substitution rate (d_s) and rate ratio (d_n/d_s) estimated by mixed evolution model from 15 sequences of HIV-1

	d_n	d_s	d_n/d_s
A - C1	0.0207	0.1178	0.176
C1 - C2	0.0214	0.0002	107
C2 - C3	0.0213	0.0002	106
C1 - C4	0.0106	0.0001	106
C4 - C5	0.0107	0.0001	107
C2 - D2	0.0681	0.0008	85
D2 - D7	0.0108	0.0001	108
D7 - D6	0.0736	0.0861	0.8549
A - E1	0.0432	0.1183	0.3652
E1 - E2	0.0216	0.0002	108
E1 - E4	0.1367	0.0111	0.0815
E1 - E5	0.0327	0.0003	109
E1 - E7	0.0835	0.0110	0.1319

phase of viral evolution process are also estimated (see Table 1). In table 1, d_n means nonsynonymous substitution rate and d_s means synonymous substitution rate, and d_n/d_s is the ratio of d_n to d_s. As mentioned above, if the viral sequences have a neutral evolution $d_n/d_s \leq 1$, otherwise the viral sequences have positively selected evolution. From Table 1, we can see that the positively selected evolution was observed between sequences C1-C2, C2-C3, C1-C4, C4-C5, C2-D2, E1-E2 and E1-E5.

4 Discussion

To understand viral evolution, it is important to investigate where the positively selected evolution occurred on the viral sequences, but that is not enough. The more important one for analysis of viral evolution is to infer when the positively selected evolution happened in the process of viral evolution. We think that the neutral evolution period and positively selected evolution period exist alternately in the process of viral evolution, and the latter is more significance to vital evolution because the viral sequences show wide-range divergence and the evolutionary rates of virus is usually accelerated in this period. The analysis of 15 sequences of HIV-1 envelope gene in this study strongly supports our hypothesis. In the first period (from year 0 to year 2), the nonsynonymous substitutions were hardly observed on the sequence A and it means the viral sequence evolved in neutrality pattern. But, in year 3, the nonsynonymous substitutions rate rapidly increased and exceeded synonymous substitution rate (d_n/d_s up to 107). It means the positive selection operated on the gene in this period. As a consequence, the viral sequence evolved into two groups, C group and E group, and a numbers of diverse sequences including quasi-species sequences (within one group) appeared in this year. Obviously, the year 3 is a positively selected evolution period. The positive selection is also observed in year 4 and year 5, and thus the another new group (D group) of viral sequence appeared in that period. But, from year 6, the nonsynonymous substitution rate rapidly reduced and the viral sequences changed to neutral evolution.

The results obtained in this study suggest that our mixed evolution model may provide more realistic description of viral evolution process than other methods assuming only neutral evolution occurs in molecular level. We think that such model could also be applied to other organism and genomes, for example, human genomes.

Mixed evolution model is still not perfect for analysis of viral evolution process that includes both neutral evolution and positively selected evolution, but, it is very significant to emphasize the necessary of consideration both neutral evolution and positively selected evolution certainly exist in evolution process when modeling molecular evolution.

References

[1] Kimura, M., Evolutionary rate at the molecular level. In *Nature* 217: 624-626, 1968.

[2] Gojobori, T., Codon substitution in evolution and the "saturation" of synonymous changes. In *Genetics* 105:1011-1027.

[3] Nei, M. and Gojobori, T., Simple methods for estimating the numbers of synonymous and nonsynonymous substitutions. In *Mol. Biol. Evol.* 3: 418- 426, 1986.

[4] Goldman, N. and Yang, Z., A Codon-based Model of Nucleotide Substitution for Protein-coding DNA Sequence. In*Molecular Biology and Evolution* 11: 725-736, 1994.

[5] Ren, F.and Tanaka, H., Analysis of Virus Evolution Based on Inhomogeneous Markov Evolution Model. In *Genome Informatics* 8: 328-329, 1997.

[6] Yang, Z. and Nielsen, R., Synonymous and Non-synonymous Rate Variation in Nuclear Genes of Mammals. In *Journal of Molecular Evolution* 46: 409-418, 1998.

[7] Holmes, E. C.*et al*, Convergent and divergent sequence evolution in the surface envelope glycoprotein of human immunodeficiency virus type 1 within a single infected patient. In *Proc. Natl. Acad. Sci. USA* 89: 4835-4839, 1992.

[8] Yang, Z., PAML: A Program for Package for Phylogenetic Analysis by Maximum Likelihood. In *CABIOS* 15: 555-556, Program web page, 1997.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Efficiencies of Information Criteria for Topology Selection in Reconstructing Molecular Phylogenetic Tree

Soichi Ogishima, Fengrong Ren and Hiroshi Tanaka
Medical Research Institute, Tokyo Medical and Dental University
1-5-45 Yushima, Bunkyo-ku, Tokyo Japan
E-mail: {ogishima@gentzen.mri.tmd.ac.jp, rencom@mri.tmd.ac.jp, tanaka@cim.tmd.ac.jp}

Abstract

In reconstruction of molecular phylogenetic tree, it has been pointed out that multifurcate phylogenetic trees are difficult to be correctly reconstructed by the conventional tree-making methods like maximum likelihood method. To resolve this problem, a new method has been developed based on information theory, which we call "model-based complexity" method. This method has been proved to be efficient in estimating multifurcate tree so far. In this study, we employed computer simulation to make further investigations about our method in the case that the exact evolution models are unknown when we reconstruct the phylogenetic tree from molecular data, and examined the results by statistical hypothesis test. The results suggest that the model-based complexity method would be superior not only to maximum likelihood method, but also to Akaike information criterion method for tree topology selection.

Keywords: model-based complexity, maximum likelihood method, Akaike information criterion, molecular phylogenetic tree

1 Introduction

In evolutionary studies, a number of methods have been proposed so far for reconstruction of phylogenetic trees from molecular data. Among these methods, maximum likelihood (ML) [1] method is considered as the most rigorous and widely used. But, it has been pointed out, even the ML method still has some problems for relevant reconstruction of evolutionary trees. One of such problems lies in the estimation of appropriate tree topology. Like other conventional methods, ML method always reconstructs fully expanded binary phylogenetic trees. But the fully expanded binary tree, namely the most complex tree is neither always true nor needed.

Two kinds of approaches are thought to be appropriate to resolve this problem. One is the approach of information criteria such as Akaike information criterion (AIC) [2] and Rissanen's minimum description length (MDL) [3] criterion by which we can deal with the tree topology selection as a sort of model selection problem. Another is to construct statistical framework to test whether a certain length of branch is 0 or not, and thus multifurcate node is detected. In our previous studies [4] [5], we have investigated the qualitative characteristics of AIC and our model-based complexity (MBC) method, which is a variant of MDL method, in estimating the appropriate tree topology. But we have not rigorously examined the effects of modeling error in base substitution.

In this study, we will take this kind of estimation error into account and strictly compare the efficiency of our MBC method with ML and AIC method in estimating the multifurcate unrooted tree by computer simulation and the results will be examined by statistical hypothesis tests.

2 Method

2.1 Model-based complexity

In the inductive inference, there would be many theories which can explain the given data to the equal extent, so that we would have to use a certain criterion to select the best one. In this context, so called "minimum complexity principle" is often used, which states that the theory which has the least complexity and nevertheless explains the data well should be chosen as the first option for true one. There are several ways to make this principle applicable to the real data. For example, in MDL principle, the complexity of data is measured by the code length of a statistical model M, $l(M)$ plus code length of data with respect to the model M, $l(D|M)$. Then minimization procedure is

taken by varying the model M within the assumed model family.

The complexity that we propose is essentially the same to the MDL principle, but the detailed structure of complexity (code length) is investigated in relation to the model inference in our formulation. Suppose we take some family of model set $M = \{M_\lambda | \lambda \in I\}$ (I is some index set of λ) which is supposed to generate data sequence D. The model-based complexity of data is defined as

$$K_{M_\lambda}(D) \equiv \inf_{M_\lambda}\{K(M_\lambda) + K(D|M_\lambda)\}, \qquad (1)$$

where $K(M_\lambda)$ is an appropriate measure of complexity defined on the model M_λ, and $K(D|M_\lambda)$ is the complexity of the residuals which the model M_λ cannot explain.

In the model-based complexity, the total complexity is composed of three terms which are (1) compositional complexity of the model, $K_c(M_\lambda)$, (2)inferential complexity of the model, $K_{in}(M_\lambda)$ and (3) empirical KL information between the data and model, $I^{KL}(D|M_\lambda)$. Then model-based complexity of data is specified as

$$K_{M_\lambda}(D) = \min_{M_\lambda}\{K_c(M_\lambda) + K_{in}(M_\lambda) + I^{KL}(D|M_\lambda)\}. \qquad (2)$$

We can use this criterion to extract the model from data by finding the model which minimizes the model-based complexity.

2.2 Complexity of evolutionary tree

In the molecular phylogenetic tree, the model space M_T is decomposed into two subspaces. One is tree model which is determined by tree topology T_p and branch lengths $\mathbf{b} = (b_1, ..., b_f)$ described either in time scale or average number of base substitutions. Another is evolution model in which the base substitution probability between two of four bases during time t is given by Markov transition matrix.

First, we describe the tree model. In evolutionary tree, the class of tree topology is determined by the number internal nodes v. In the case of unrooted tree, when $v = \epsilon - 2$ (ϵ is the number of leaves), we have a fully expanded binary tree. The complexity of natural number v is given by $\log^* v$, where $\log^* v = \log v + \log\log v + \log\log\log v + \ldots$ [3]

Even if the number of internal nodes v is determined, the tree is not unique. In the case of unrooted trees, the possible number of the tree topologies are given by $\frac{1}{f}\binom{\epsilon + v - 2}{v}$ so the resultant complexity is given

$$K_c(v) = \log^* v + \log\frac{1}{f}\binom{\epsilon + v - 2}{v}. \qquad (3)$$

The complexity of branch lengths which are considered as inferential parameters is given by

$$K_{in}(\mathbf{b}) = \frac{\epsilon - \dim(\mathbf{b})}{2}\log n, \qquad (4)$$

where n which describes the number of data is now given by the length of nucleotide sequence, and $\epsilon - \dim(\mathbf{b})$ is the number of efficient eigen vectors of Fisher's information matrix $I^F(\mathbf{b})$.

Second, we describe the evolution model. We used HKY model for rate matrix $\mathbf{R}$ in which the difference of transition (α) and transversion (β) base substitution rate and the frequency of each base (π_i) are taken into account. The details would be referred elsewhere [6]. We use this transition matrix to construct the whole likelihood of the molecular sequences $s_1, s_2, \cdots, s_\epsilon$ when the tree topology is assumed to be M_T. The procedure to construct the likelihood function is same as the Felsenstein's ML method. Taking the sum of the complexities of the each component of a phylogenetic tree, we have the following total complexity of phylogenetic tree to be minimized:

$$\begin{aligned} K_{M_T}(s_1, s_2, ..., s_\epsilon) &= \min_{v,\mathbf{b}}\Bigg\{ -\log L(s_1, s_2, \cdots, s_\epsilon | M_T) \\ &+ \left[\log^* v + \log\frac{1}{f}\binom{\epsilon + v - 2}{v}\right] \\ &+ \frac{\epsilon - \dim(\mathbf{b})}{2}\log n\Bigg\}. \end{aligned} \qquad (5)$$

Where $\log L(s_1, s_2, \cdots, s_\epsilon | M_T)$ is log likelihood function which approximates $I^{KL}(D|M_T)$.

2.3 Computer simulation and statistical hypothesis tests

We take a tree with 4 OTU's as the topology model to be compared. The length of branch c in Figure 1 is varied from 0.0 to 0.1 to realize the bifurcate and multifurcate tree. That is, if $c = 0$ then the topology becomes multifurcate tree; if c is relatively small comparing with a and b then the topology will be nearly multifurcate, whereas if c is statistically significantly large then the tree becomes definitely binary. Two group's data which are assumed to be 1000bp and 3000bp respectively are generated

Three criteria, maximum likelihood, Akaike information criterion and model-based complexity are applied to this simulation to select whether the trees

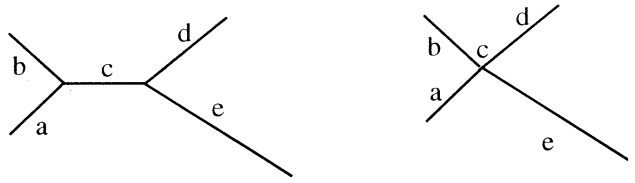


Figure 1: Bifurcate and multifurcate tree models for this study

are bifurcate or multifurcate. Because the negative logarithm of likelihood is equal to Shannon's self-information, ML also can be considered as a kind of information criterion. These three methods estimate parameter values by maximizing the same likelihood function, but as the additional complexity terms are different, so that they make different selection for tree topology. We investigate the changes of the frequencies in selecting tree topology when the true branch length c varies. As at some c value, the frequencies to select bifurcate or multifurcate tree will become equal, we denote this value of c by $c_{ML}(0.5)$, $c_{AIC}(0.5)$, $c_{MBC}(0.5)$ and call it equiprobable (EP) value.

The results by three complexity criteria are examined by using statistical hypothesis test. **Null hypothesis** H_0: the branch length c is 0 and the corresponding node is multifurcate. **Alternative hypothesis** H_A: the branch length c is not 0 and the corresponding node is bifurcate. With the hypotheses of H_0 and H_A, the statistical test framework is as follows:

(1) Null hypothesis H_0 is rejected and H_A is accepted if the estimation c exceeds the threshold $\theta_c(\alpha)$, which means the node is bifurcate.

(2) Null hypothesis H_0 is not rejected but accepted if the estimation c is below the threshold $\theta_c(\alpha)$, which means that the node is multifurcate.

$\theta_c(\alpha)$ is the value which satisfies the probability $P(c > \theta_c(\alpha)) = \alpha$, and the value α is significant level in statistical test (Type I error). In order to determine $\theta_c(\alpha)$, we should calculate the probabilistic distribution of c when the true value of c is 0. In this simulation, we apply the ML method to the simulated sequence data that are generated under the assumption of $c = 0$ to obtain the distribution of c. The decision rule for topology selection is whether estimated value c is greater than the threshold $\theta_c(\alpha)$ or not. Ordinarily the value α is taken to be 0.05 or 0.01.

We also examine the probability of type II error. It is the probability of error that H_0 is not rejected though H_A is correct. The probability of type II error is denoted by β. $1 - \beta$ is considered as the power of statistical test. The value $c^{\alpha}_{SS}(0.5)$ denotes the true c value in which the estimation of type II error (β) becomes 0.5. This value characterizes the global nature of this statistical test to detect multifurcation and has the same role as the EP value that defined in the above information criterion methods.

3 Results

The changes of the frequencies in selecting a bifurcate or multifurcate tree when the branch length c varies are shown in Figure 2. We can see that the ML method is almost always to select a bifurcate tree except few cases. Even if the true $c = 0$ (exactly multifurcate), the ML method still selects the bifurcate tree by about 30% of the cases. On the contrary, AIC, MBC and simulation statistics method show more natural tendencies that fit our intuition. Thus, we only compare the results obtained by AIC, MBC and simulation statistics method in the following.

Table 1 shows the results when $\kappa(= \alpha/\beta)$ is fixed. In the case that simulated sequences are assumed to be 1000bp, the AIC method considers the tree as multifurcate until the c value is varied up to 0.024 ($0 \leq c \leq 0.024$). The MBC method is more conservative in selection of a bifurcate tree, because the topology is not considered as bifurcate until the c value is varied up to 0.051 ($0 \leq c \leq 0.051$). On the other hand, in the simulation statistics method, the threshold $c^{0.05}_{SS}(0.5)$ in this case is 0.021. This threshold 0.021 corresponds well with AIC EP value, 0.024, whereas the MBC EP value, 0.051, exceeds even 99% threshold $c^{0.01}_{SS}(0.5)$, 0.029. Therefore, if we consider the simulation statistics method is reliable, the AIC method appears to be relatively good method for selection between the bifurcate and multifurcate tree topology and the MBC method seems to be biased to the multifurcate tree topology. The same tendency is observed in the case that the length of simulated sequence is assumed to be 3000bp.

Table 2 shows the results when κ is varied to take the estimation errors of κ into account. In this case, the thresholds $c^{0.05}_{SS}(0.5)$ is 0.026 and it is still correspond well with AIC EP values, 0.023. But the threshold $c^{0.01}_{SS}(0.5)$, 0.041, corresponds well with MBC EP

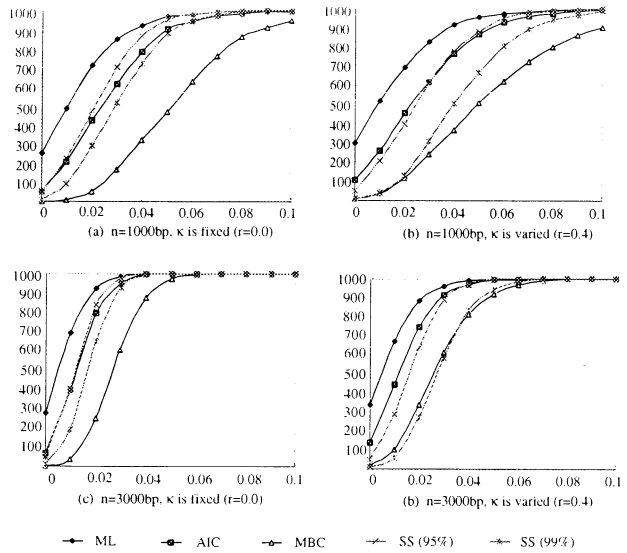


Figure 2: Changes of the selection ratio among the two topologies obtained from different methods. Ordinate denotes the frequencies that the bifurcate trees are selected among1000 samples. Abscissa denotes the true c values used in generation of the data.

value, 0.049. Especially, when the sequence length increase to 3000bp, the MBC EP value, 0.026, corresponds very well with threshold $c_{SS}^{0.01}(0.5)$, 0.029.

4 Discussion

In this study, the effects of estimation errors of evolution model are taken into account by varying κ in reconstruction of phylogenetic trees including multifurcation, and the efficiencies of the AIC and MBC methods are investigated in light of the results obtained by simulation statistics approach. In the case that the evolution model is exactly known, AIC method appears to be relatively good method for selection between the bifurcation and multifurcation, and MBC method seems to be biased to the multifurcate topology. However, in the case that evolution model is not exactly estimated, the MBC method shows good performance in selection between the bifurcation and multifurcation, and the AIC method seems to be biased to the bifurcate topology. In realistic world, the exact evolution model of the species is usually unknown, thus we have to estimate the model from molecular sequence data. Considering this situation, we would say that our MBC method provides fairy robust criterion for tree topology selection compared with ML and AIC method, because the AIC method, needless to say ML method, overestimates the complexity of tree topology to prefer the bifurcate tree. Furthermore, with the increase of the sequence length (over 3000bp), the tendency of underestimation is improved and the MBC method becomes more efficiency. These results suggest that our MBC method would provide a better criterion than ML and AIC methods for selecting tree topology.

Table 1: The results when κ is fixed ($r = 0.0$)

	$c_{AIC}(0.5)$	$c_{MBC}(0.5)$	$c_{SS}^{0.05}(0.5)$	$c_{SS}^{0.01}(0.5)$
1000bp	0.024	0.051	0.021	0.029
3000bp	0.012	0.027	0.012	0.017

Table 2: The results when κ is varied ($r = 0.4$)

	$c_{AIC}(0.5)$	$c_{MBC}(0.5)$	$c_{SS}^{0.05}(0.5)$	$c_{SS}^{0.01}(0.5)$
1000bp	0.023	0.049	0.026	0.041
3000bp	0.011	0.026	0.017	0.029

References

[1] J.Felsenstein, Evolutionary Trees from DNA Sequences: a Maximum Likelihood Approach. *J. Mol. Evol.*, 17, 368-376 (1981).

[2] H.Akaike and H.Hirotugu, A New Look at the Statistical Model Identification. *IEEE Trans. Automatic Control*, 19(6), 716-723 (1974).

[3] J. Rissanen, Modeling by shortest data description. *Automatica*, 14, 465-471(1978).

[4] F. Ren, H. Tanaka and T. Gojobori, Construction of Molecular Evolutionary Phylogenetic Tree from DNA Sequences Based on Minimum Complexity Principle. *Computer Methods and Programs in Biomedicine*, 46, 121-130(1995).

[5] H.Tanaka, F.Ren, T.Okayama and T.Gojobori, Inference of Molecular Phylogenetic Tree Based on Minimum Model Based Complexity Method. *Proc. 5th intl.conf. on Intelligent Systems for Molecular Biology*, 319-328(1997).

[6] M.Hasegawa, H.Kishino and T.Yano, Dating of the Human-ape Splitting by a Molecular Clock of Mitochondrial DNA. *J. Mol. Evol.*, 22, 160-174 (1985).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

On a Model of Artificial Cell

Yasuhiro Suzuki and Hiroshi Tanaka
Medical Research Institute, Tokyo Medical and Dental University
1-5-45 Yushima, Bunkyo-ku Tokyo Japan
Email: {suzuki@gentzen.mri.tmd.ac.jp, tanaka@tmd.ac.jp }

Abstract

Abstract

We develop an Artificial Cell System (ACS) which corresponds to a class of P system. ACS consists of a multiset of symbols, a set of rewriting rules (reaction rules) and membranes. We implemented the system by using LISP. Throughout a simulation, we find that cells evolve to a structure consists of several cell-like membranes, recurrently placed under certain condition. We investigate the correlation between the type of reaction rules and evolving of a cell and find an parameter to describe the correlation.

Keywords: Artificial Chemistry, Artificial Life, Membrane Computing

1 Introduction

Membrane is irreducible structure for living systems. It distinguishes "*self*" from its environment and it composes hierarchical structures (cells, organs and so on). Membranes change their structure dynamically. We are interested in their dynamical structure. Hence, we develop an "*Artificial Cell System* (*ACS* [4])." It consists of a multiset of symbols, a set of rewriting rules (*reaction rules*) and membranes. A membranes is composed of "chemical compounds (in fact they are symbols) " that are generated throughout chemical reactions in the cell. In each cell there are some chemical compounds and these chemical compounds interact with each other according to the reaction rules.

ACS is composed on an abstract chemical system, "*Abstract Rewriting System on MultiSets (ARMS)*" that is multiset transform system we have studied [4]. It is consisted of a multiset of symbols and a set of reaction rules.

Since *ARMS* is not a strings rewriting system but (multi)-*set* rewriting system, it can deal with many degrees of freedom such as *chemical solution* and the *concentration* of a *chemical compounds*. We confirmed that it can simulate phenomenon that we find in real bio-chemical reactions such as "*Belouzov-Zhabotinsky reaction (BZ-reaction) [4]*" (*BZ-reaction* is spontaneous chemical oscillation, it is considered as one of a basic mechanisms of bio-chemical system).

P systems

ACS corresponds to a class of P systems [3]. P systems that is proposed by G. Păun is a parallel molecular computing model based on processing multisets of object in cell-like membrane structures. P systems is a multiset transformation system, however, it is different from [2], it introduces "*membrane*" to its computing mechanism. In the system *cells* are structured like a living cell. It is composed of several cells that are delimited from the neighboring cells, that evolve internally. Technically, it is structured like a Venn diagram, in the diagram each set corresponds to a multiset. With all the considered sets being subsets of a unique set, this set corresponds to "*skin*" membrane, and not allowed to be intersected each other.

In the system, objects in each cell evolve with specified evolution rules for objects. Any object, alone or together with one or more object, evolves, can transformed in other objects, can pass through one membrane, and can dissolve the membrane in which it is placed. All objects evolve at the same time, in parallel; in turn, all membranes are active in parallel.

The computing power of P systems are equal to Turing machine was proved and an algorithm to computer SAT problem in linear time was proposed. There have already proposed various P systems and their mathematical properties have investigated other than these [3].

We implemented *ACS*, a class of P system, by using LISP. We do not implement the system biochemically. The reason why we use "LISP" in the title is to emphasize it. We will not address the implementation of

especially when using LISP but we propose a method of implementation.

2 *ARMS*

We will introduce multiset rewriting system, "*Abstract Rewriting system on MultiSets*" in this section. Intuitively, *ARMS* is like a chemical solution in which floating *molecules* can interact with each other according to reaction rules. Technically, a chemical solution is a finite multi-set of elements denoted $A^k = \{a, b, \ldots, \}$; these elements correspond to *molecules*, and reaction rules are specified in *ARMS* of rewrite rules. As to the intuitive meaning of an *ARMS*, we refer to the study of chemical abstract machines [2]. In fact, this system can be thought of as reflecting an underling "*algorithmic chemistry* [1]."

Let A be an *alphabet* (a finite set of abstract symbols). The set of all strings over A is denoted by A^*; the empty string is denoted by λ. (Thus, A^* is the free monoid generated by A under the operation of concatenation and having the identify λ.) The length of a string $w \in A^*$ is denoted by $|w|$.

A *rewriting rule* over A is a pair of strings (u, v), $u, v \in A$. We write such a rule in the form $u \rightarrow v$. Note that u and v can also be empty. A *rewriting system* is a pair (A, R), where A is an alphabet and R is a finite set of rewriting rules over A.

With respect to a rewriting system $\gamma = (A, R)$ we define over A^* a relation $\Longrightarrow$ as follows: $x \Longrightarrow y$ iff $x = x_1 u x_2$ and $y = y_1 v x_2$, for some $x_1, x_2 \in A^*$ and $u \rightarrow v \in R$. The reflexive and transitive closure of this relation is denoted by $\Longrightarrow^*$. A string $x \in A^*$ for which there is no string $y \in A^*$ such that $x \Longrightarrow y$ is said to be an *dead* one (it cannot evolve by means of the rewriting rules).

From now on, we work with an alphabet A whose elements are called 1 *objects*; the alphabet itself is called a *set of objects*.

A *multiset* over a set of objects A is a mapping $M : A \longrightarrow \mathbf{N}$, where $\mathbf{N}$ is the set of natural numbers, 0, 1, 2,.... The number $M(a)$, for $a \in A$, is the *multiplicity* of object a in the multiset M. Note that we do not accept here an infinite multiplicity. The set $\{a \in A \mid M(a) > 0\}$ is denoted by $supp(M)$ and is called the *support* of M. The number $\sum_{a \in A} M(a)$ is denoted by $weight(M)$ and is called the *weight* of M.

We denote by $A^\#$ the set of all multisets over A, including the empty multiset, $\emptyset$, defined by $\emptyset(a) = 0$ for all $a \in A$.

A multiset $M : A \longrightarrow \mathbf{N}$, for $A = \{a_1, \ldots, a_n\}$, can be naturally represented by the string $a_1^{M(a_1)} a_2^{M(a_2)} \ldots a_n^{M(a_n)}$ and by any other permutation of this string. Conversely, with any string w over A we can associate a multiset: denote by $|w|_{a_i}$ the number of occurrences of object a_i in w, $1 \leq i \leq n$; then, the multiset associated with w, denoted by M_w, is defined by $M_w(a_i) = |w|_{a_i}, 1 \leq i \leq n$.

The union of two multisets $M_1, M_2 : A \longrightarrow \mathbf{N}$ is the multiset $(M_1 \cup M_2) : A \longrightarrow \mathbf{N}$ defined by $(M_1 \cup M_2)(a) = M_1(a) + M_2(a)$, for all $a \in A$. If $M_1(a) \leq M_2(a)$ for all $a \in A$, then we say that multiset M_1 is included in multiset M_2 and we write $M_1 \subseteq M_2$. In such a case, we define the multiset difference $M_1 - M_2$ by $(M_2 - M_1)(a) = M_2(a) - M_1(a)$, for all $a \in A$. (Note that when M_1 is not a included in M_2, the difference is not defined).

A rewriting rule such as

$$a \rightarrow a \ldots b,$$

is called a *heating rule* and denoted as $r_{\Delta > 0}$; it is intended to contribute to the stirring solution. It breaks a complex *molecule* into smaller ones: *ions*. On the other hand, a rule such as

$$a \ldots c \rightarrow b,$$

is called a *cooling rule* and denoted as $r_{\Delta < 0}$; it rebuilds *molecules* from smaller ones. In this paper, reversible reactions, i.e., $S \rightleftharpoons T$, are not considered. We shall not formally introduce the refinement of *ions* and *molecules* though we use refinement informally to help intuition (on both types of rules we referred to [2]).

A *multiset rewriting rule* (we also use to say, *evolution rule*) over a set A of objects is a pair (M_1, M_2), of elements in $A^\#$ (which can be represented as a rewriting rule $w_1 \rightarrow w_2$, for two strings $w_1, w_2 \in A^*$ such that $M_{w_1} = M_1$ and $M_{w_2} = M_2$). We use to represent such a rule in the form $M_1 \rightarrow M_2$.

3 Artificial Cell System

We introduce the basic structural ingredient, membrane structures to *ARMS*.

3.1 Membrane structure

We will introduce membrane and its structure to *ARMS* with referring [3]. Firstly, we will define the language MS over the alphabet $\{[,]\}$, whose strings are recurrently defined as follows:

(1) $[,] \in MS$

(2) if $\mu_1, ..., \mu_n \in MS$, n ≥ 1, then $[\mu_1, ..., \mu_n] \in MS$;

(3) nothing else in MS.

Consider now the following relation on MS: for $x, y \in MS$ we write $x \sim y$ if and only if we can write the two strings in the form $x = [_1...[_2...]_2[_3...]_3...]_1, y = [_1...[_3...]_3[_2...]_2...]$ (two pairs of parentheses which are not one contained in the other are interchanged, together with their contains). We also denote by $\sim$ the reflexive and transitive closures of the relation $\sim$. This is clearly and equivalence relation. We denote by $\overline{MS}$ the set of equivalence classes of MS with respect to this relation. The elements of $\overline{MS}$ are called *membrane structures.*

It is easy to see that the parentheses [,] appearing in a membrane structure are correctly matching, in the usual sense. Conversely, any string of correctly matching pairs of parentheses [,], with a matching pair at the ends, corresponds to a membrane structure.

Each matching pair of parentheses [,] appearing in a membrane structure is called a *membrane.* The number of membranes in a membrane structure μ is called the *degree* of μ and denoted by deg(μ). The external membrane of a membrane structure μ is called the *skin* membrane of μ. When a membrane which appears in $\mu \in \overline{MS}$ in the form [ ] and no other membrane appearers inside the two parentheses then it is called an *elementary* membrane.

3.2 Dissolving and dividing membrane

A membrane is composed of a "membrane compound (in fact, it is a symbol)." To maintain a membrane, it needs a certain volume of it. A membrane disappear if the volume of membrane compounds decreases under the needed volume to maintain the membrane. Dissolving membrane is defined as follows;

$$[_h a, ...[_i b, ...]_i]_h \rightarrow [_h a, b, ...]_h,$$

where {...} illustrates chemical compounds inside a membrane. Dissolving is caused when

$$\frac{M(membrane\ compounds)}{|M|} < \delta,$$

where δ is slash-hold value of dissolving membrane. A membrane is dissolved then all chemical compounds in its region are left free in the region and they are merged into the region immediately above it.

On the other hand, when the volume of membrane compounds increases to a certain extend, a membrane is divided. Dividing membrane is realized as dividing a multiset in the random size. The frequency of dividing membrane is decided in proportion to its size. As the size of a multiset becomes larger, the cell is divided frequently. Technically, it is defined as follows;

$$[_h a, b, ...]_h \rightarrow [_i a, ...]_i, [_j b, ...]_j$$

Dividing is caused when

$$\frac{M(membrane\ compounds)}{|M|} < \sigma,$$

where σ is slash-hold value of dividing membrane. A membrane is divided then all chemical compounds in its region are left free in the region and they are separated randomly by new membranes.

λ_e parameter and system's behavior In order to investigate the correlation between the reaction rule and the behavior of the model, we will introduce the λ_e parameter [4]. This parameter is introduced as an order parameter of $ARMS$.

Let us define the λ_e parameter as follows:

$$\lambda_e = \frac{\Sigma r_{\Delta S>0}}{1 + (\Sigma r_{\Delta S<0} - 1)} \quad (1)$$

where $\Sigma r_{\Delta S>0}$ corresponds to the number of *heating rules*, and $\Sigma r_{\Delta S<0}$ to the number of *cooling rules.* This parameter is defined when the number of rules is greater than 1.

When the $ARMS$ only uses rules of the type $r_{\Delta S<0}$, λ_e is equal to 0.0. On the contrary, if the $ARMS$ uses rules of the type $r_{\Delta S>0}$ and $r_{\Delta S<0}$ with the same frequency, λ_e is equal to 1.0. Finally, when the $ARMS$ only uses rules of the type $r_{\Delta S>0}$, λ_e is greater than 1.0.

The λ_e indicates the degree of reproduction in a cell. When the λ_e closes to 0.0, the degree is quiet low and λ_e is away from 1.0, the degree becomes high, vice versa.

Behavior of ACS is classified into four classes by this parameter as follows;

- Type I: A cell dose not evolve and diminish;
- Type II: "*Cell cycle*" like behavior, the period of dividing cell emerges cyclically, can be seen.
- Type III: A cell evolves to a complex, hierarchically structured cell;
- Type IV: All chemical compounds inside a cell increase rapidly, however, the cell is hardly to be divided;

where each λ_e value is not important very much. Although they are changed under the different environments, these four classes are unchanged.

Type I (λ_e closes to 0.0)

When the λ_e close to 0.0 the *cooling rule* is mainly used, thus cells are hardly to grow up. Thus membranes hardly to be divided.

Type II (λ_e in between 0.5 and $\sim$ 1.0)

When the λ_e in between $0.5 \sim 1.0$, membranes is likely to be divided than λ_e is close to 0.0. But, since *cooling rules* are likely to be used, the membrane compounds do not increase very much. Thus, when a cell evolves a certain size the membrane compounds of each cell decreases and they are dissolved. As a while, "*cell cycle*" like behavior, period of dividing cell emerges cyclically, can be seen in this range.

Type III (λ_e in between 1.05 $\sim$ 2.33)

When the λ_e exceeds 1.0 and *heating rule* is likely to be used, a cell grows up easier and the frequency of dividing cell becomes high. Thus a cell evolves a large complex cell.

Type IV (λ_e more than 2.5)

The λ_e parameter becomes much larger then 1.0, membranes hardly to be divided. Because, the number of all kinds of compounds in the cell increase and it is difficult to especially the number of only membrane compounds increase then a cell is hardly to be divided.

We believe that these four classification describe the behavior of the system basically. But, when δ or σ is changed the λ_e value that corresponds to each type is also changed. Thus it needs to investigate the correlation among δ, σ and λ_e more precisely.

4 Conclusion, to beyond Turing Machine

A. Turing abstracts *computation* in terms of calculation by *pen (ink)* and *paper*; it is Turing Machine. Many mathematician have tried to beyond Turing Machine's computational power.

It is obvious that living systems never do calculation only by using *pen* and *paper*. Thus, if we could abstract *computation* in terms of living systems do then we would obtain new computational world.

We have not found the new world yet, however, we have already obtained some *hints* throughout observation of living system. One of it is "parallelism." In a living system, every cell and every organ is acting in parallel. Thus, if some organs or cells lose activity then as a whole, another organs or cells try to increase its activity for recovering and the system keep its activity. S. Rasmussen proposed us to implement membrane computing system on a parallel computer. *ACS* is an abstraction of computation of living systems thus parallelism is an irreplaceable concept.

Implementing our system on a parallel computer and to investigate its mathematical property are our future work. We expected that these studies lead us to a new computing paradigm beyond Turing Machine.

Acknowledgment

The authors would like to express many thanks to Dr. Gheorghe Păun for his useful comments, discussion, mathematical refinements. This research is supported by Grants-in Aid for Scientific Research No.11837005 from the Ministry of Education, Science and Culture in Japan.

References

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

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

[3] Păun, G. 1998. Computing with Membranes, *Turku Center for Computer Science TUCS Technical Report No. 208* (submitted, on also http://www.tucs.fi)

[4] Suzuki, Y. and H. Tanaka. 1998. Order parameter for a Symbolic Chemical System, *Artificial Life VI*:130-139, MIT press.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Propagation of Stochastic Signals in Probabilistic Universal Learning Networks and Validation of Its Calculation Method

Chunzhi Jin, Kotaro Hirasawa, Murata Junichi and Jinglu Hu

Dept. of Electrical and Electronic Systems Engineering, Kyushu University,
6-10-1 Hakozaki-ku, Fukuoka, 812-8581, Japan.
Tel: 092-642-3986, Fax: 092-642-3962
E-mail: jin@ees.kyushu-u.ac.jp

Abstract

Probabilistic Universal Learning Network (PrULN) is a learning network for dealing with stochastic signals. In this paper, error bounds of mean and covariance are presented which is used to estimate approximation error of mean and covariance of node outputs in PrULN. Through a discussion on the error term, it is concluded that choice of the node functions having small value of higher order derivatives, choice of small initial values of learning parameters and normalization of input signals are favorable for reducing the error of approximate calculation.

1 Introduction

Learning networks are widely used to the fields such as pattern recognition, system identification and system control[1]. When real world data are dealt with, it is must be considered that noise is being added. It is very significant to investigate what statistic feature the node outputs of the network has when stochastic signals are inputted to the network, and to develop a learning algorithm to make the node outputs of the network obtain the desired stochastic feature. Probabilistic Universal Learning Network (PrULN)[2] was precisely devised to deal with the stochastic signals positively.

Generally, it is expected that the system is strong to noise in the fields of signal processing and system control. PrULN is an extension of the usual learning network in terms of dealing with the stochastic signals. In PrULN scheme, the means and covariances of the node outputs can be calculated from the moment values of the inputs of network, and furthermore, the network can be trained so that the node outputs have desired stochastic feature. The basic idea of PrULN and some application examples have been described in the literature[2, 3, 4]. But, further investigation about signal propagation and network construction is needed, especially, the stochastic characteristic values of the node outputs are approximately calculated in PrULN, so it is important to establish a method for appropriately estimating the error in the sense that an appropriate network structure is selected.

In this paper, a calculation method of the error bound is proposed which is used to estimate the error due to approximate calculation of each moment value of the node outputs in PrULN. The error bound equations are derived for both of general case and a case in which the noise is white. In addition, through a discussion on the error bound, some hints about decision of node activation function and setting of the network initial parameters are presented. In Section 2, PrULN is described briefly. In Section 3, calculation equations of the means and covariances of node outputs in the network and the calculation equations of the error bound are derived, and the meanings of the equations are discussed. Section 5 summarizes conclusions.

2 PrULN

A Probabilistic Universal Learning Network is a learning network equipped with a capability of deal with stochastic signals. It has a similar structure to deterministic one, but it propagates stochastic signals and controls both means and covariances of propagating signal.

PrULN consists of two kinds of element – nodes and branches. Nodes are assigned appropriate node functions including a function realized by a subnetwork, *e.g.,* a sigmoid neural network or a neurofuzzy network, and are interconnected with multi-branches that have arbitrary – positive, zero, or even negative – time delays. PrULN has sufficient flexibility in the structure, one can take this advantage to incorporate a priori knowledge about systems as much as possible.

Figure 1 shows PrULN architecture. It is clear that when sigmoid node function are used in all nodes, the PrULN reduces to one with similar structure to a common sigmoid

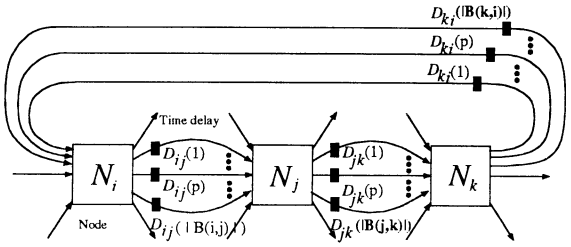


Figure 1: Structure of Universal learning networks.

neural network. The PrULN dynamics is described as follows

$$h_j(t) = f_j(\{h_i(t - D_{ij}(p)) \mid i \in JF(j), p \in B(i,j)\}, \{r_n(t) \mid n \in N(j)\}, \{\lambda_m(t) \mid m \in M(j)\}), \quad (j \in J,\ t \in T), \tag{1}$$

where $h_j(t)$ is the output value of node j at time t, $r_n(t)$ the value of n-th external input variable at time t, $\lambda_m(t)$ the value of m-th parameter at time t, f_j the nonlinear function of node j, $D_{ij}(p)$ the time delay of p-th branch from node i to node j, J the set of suffixes of nodes $\{j\}$, $JF(j)$ the set of suffixes of nodes which are connected to node j, $JB(j)$ the set of suffixes of nodes which are connected from node j, N the set of suffixes of external input variables $\{n\}$, $N(j)$ the $\{n|r_n$ is fed to node $j\}$, M the set of suffixes of parameters $\{m\}$, $M(j)$ the $\{m|h_j$ is partially differentiable with respect to $\lambda_m\}$, $B(i,j)$ the set of suffixes of branches from node i to node j, T the discrete set of sampling instants, respectively.

A PrULN has learning ability, provided with adjustable parameters $\lambda_m(t)$. The criteria function used in learning contains not only the term of mean square error of the node outputs, but also a term which make the statistic values of the node outputs close to desired values. As explained in detail below, higher information than second order derivative is necessary to calculate the means and covariances of the node outputs, so more higher order derivatives have to be calculated, when gradient based learning algorithm is used to optimize the criteria function including means and covariances term. For PrULN a generalized learning algorithm is revised which is based on forward or backward propagation of higher ordered derivatives.

When PrULNs are applied to system control, the higher order derivatives of a criteria function with respect to the parameters of controller are systematically calculated by forward or backward propagation. In other words, both of the control object and the controller are considered as one network, and higher order derivative of the node outputs $h_j(t)$ with respect to the parameters λ_m are calculated, then the training based on the gradient is performed. So an optimized controller is realized.

3 Approximate calculation of means and covariances, and their errors

In PrULN, a node output $h_p(t)$ on the instant t is determined by the network initial values and the external inputs inserted into the system until the instant t. From (1) we can see that the node output is related to many variables. For the convenience, let us pay attention to the probabilistic variables only, and list them as $z_1, z_1, \cdots, z_\ell$, then the node output $y(t)$ can be rewritten by the following simple form

$$h_p(t) = f_p(z_1, z_2, \cdots, z_\ell). \tag{2}$$

From now, several equations for calculating means and covariances of node outputs and for estimating the calculation error are derived. And also, a discussion about the calculation error is presented. Where, mean, variance, covariance and n–th central moment are denoted by $E[\cdot]$, $Var[\cdot]$, $Cov[\cdot]$, $\mu_n[\cdot]$, respectively.

3.1 Means

The mean of $h_p(t)$ is defined as its ensemble average $E[h_p(t)]$. By the multi-variable Taylor's Theorem, expanding (2) around the point of $(E[z_1], E[z_2], \cdots, E[z_\ell])$ up to second-order term, we have

$$\begin{aligned} h_p(t) &= f_p(E[z_1], E[z_2], \cdots, E[z_\ell]) \\ &+ \sum_i \frac{\partial^+ h_p(t)}{\partial z_i}\Big|_E \delta_i \\ &+ \frac{1}{2}\sum_{i,j} \frac{\partial^{+2} h_p(t)}{\partial z_i \partial z_j}\Big|_E \delta_i \delta_j \\ &+ \Delta_{h3}, \end{aligned} \tag{3}$$

where $\frac{\partial^+ h_p(t)}{\partial z_i}|_E$ and $\frac{\partial^{+2} h_p(t)}{\partial z_i \partial z_j}|_E$ are ordered derivatives evaluated at $(E[z_1], E[z_2], \cdots, E[z_\ell])$, $\delta_i = (z_i - E[z_i])$ and Δ_{h3} is remained term of second expansion

$$\Delta_{h3} = \frac{1}{3!}\sum_{i,j,k} \frac{\partial^{+3} h_p(t)}{\partial z_i \partial z_j \partial z_k}\Big|_\theta \delta_i \delta_j \delta_k, \tag{4}$$

where derivative $\frac{\partial^{+3} h_p(t)}{\partial z_i \partial z_j \partial z_k}|_\theta$ is evaluated at a point $(E[z_1] + \theta(z_1 - E[z_1]),\ E[z_2] + \theta(z_2 - E[z_2]),\ \cdots,\ E[z_\ell] + \theta(z_\ell - E[z_\ell]))$, θ is an appropriate value being in (0, 1). Taking ensemble average to (3), mean of $h_p(t)$ is obtained as follows

$$\begin{aligned} E[h_p(t)] &= f_p(E[z_1], E[z_2], \cdots, E[z_\ell]) \\ &+ \frac{1}{2}\sum_{i,j} \frac{\partial^{+2} h_p(t)}{\partial z_i \partial z_j}\Big|_E Cov[z_i, z_j] + e_m, \end{aligned} \tag{5}$$

e_m is error term of approximate calculation of mean value $E[h_p(t)]$,

$$e_m = E[\Delta_{h3}]. \tag{6}$$

By absolute inequality, we have

$$|e_m| \leq \frac{1}{3!}\sum_{i,j,k}\left|\frac{\partial^{+3}h_p(t)}{\partial z_i\partial z_j\partial z_k}\right|_{max}\cdot\left|\mu_3[z_i,z_j,z_k]\right|, \tag{7}$$

where $|\frac{\partial^{+3}h_p(t)}{\partial z_i\partial z_j\partial z_k}|_{max}$ is maximum of $|\frac{\partial^{+3}h_p(t)}{\partial z_i\partial z_j\partial z_k}|_\theta$. Right side of (7) is an error bound of e_m.

In many engineering problems the noises are considered as white Gaussian. Now, we assume that $z_1, z_2, \cdots, z_\ell$ are white Gaussian the expectations are zero

$$E[z_i] = 0, \quad (i = 1,\cdots,\ell),$$

and the variances are σ_i^2

$$Var[z_i] = E[z_i^2] = \sigma_i^2 \ \ (\sigma_i > 0), \quad (i = 1,\cdots,\ell)$$

so, each central moment of z_i can be calculated as bellows

$$\mu_n[\underbrace{z_i,\cdots,z_i}_{n}] = E[z_i^n] = \begin{cases} 0, & n \text{ is odd} \\ 1\cdot 3\cdot 5\cdots(n-1)\sigma_i^n, & n \text{ is even } \&\ \ (n>1), \end{cases} \tag{8}$$

and when those variables are independent each other in the sense of probability, covariance $Cov[z_i,z_j]$ $(i \neq j)$ become to zero, so mean of p–th node output can be simply calculated by the following equation

$$\begin{aligned} E[h_p(t)] &= f_p(E[z_1],E[z_2],\cdots,E[z_\ell]) \\ &\quad + \frac{1}{2}\sum_i \frac{\partial^{+2}h_p(t)}{\partial z_i^2}\Big|_E Var[z_i] + e_m \qquad (9) \\ &= f_p(0,0,\cdots,0) \\ &\quad + \frac{1}{2}\sum_i \frac{\partial^{+2}h_p(t)}{\partial z_i^2}\Big|_E \sigma_i^2. \qquad (10) \end{aligned}$$

where the error e_m is equal to zero because $\mu_3[z_i,z_j,z_k] = 0$ in this case (See (7)).

3.2 Covariances

Using equation (3) and (5), the covariances of $y_p(t)$ and $y_q(t)$, $Cov[y_p,y_q]$ can be derived as follows

$$\begin{aligned} &Cov[y_p(t),y_q(t)] \\ &= E[(y_p(t)-E[y_p(t)])(y_q(t)-E[y_q(t)])] \\ &= \sum_{i,\ell}\frac{\partial^+h_p(t)}{\partial z_i}\Big|_E\frac{\partial^+h_q(t)}{\partial z_\ell}\Big|_E Cov[z_i,z_\ell] \\ &\quad + \frac{1}{2}\sum_{i,j,\ell}\frac{\partial^{+2}h_p(t)}{\partial z_i\partial z_j}\Big|_E\frac{\partial^+h_q(t)}{\partial z_\ell}\Big|_E\cdot\mu_3[z_i,z_j,z_\ell] \\ &\quad + \frac{1}{2}\sum_{i,\ell,m}\frac{\partial^+h_p(t)}{\partial z_i}\Big|_E\frac{\partial^{+2}h_q(t)}{\partial z_\ell\partial z_m}\Big|_E\cdot\mu_3[z_i,z_\ell,z_m] \\ &\quad + e_c, \end{aligned} \tag{11}$$

where e_c is approximate error of covariance. By straightforward calculation, we have

$$\begin{aligned} &e_c \\ &= \frac{1}{4}\sum_{i,j,\ell,m}\frac{\partial^{+2}h_p(t)}{\partial z_i\partial z_j}\Big|_E\frac{\partial^{+2}h_q(t)}{\partial z_\ell\partial z_m}\Big|_E \\ &\quad\cdot(\mu_4[z_i,z_j,z_\ell,z_m] - Cov[z_i,z_j]Cov[z_\ell,z_m]) \\ &+ \frac{1}{3!}\sum_{i,j,k,\ell}\frac{\partial^+h_q(t)}{\partial z_\ell}\Big|_E E\Big[\frac{\partial^{+3}h_p(t)}{\partial z_i\partial z_j\partial z_k}\Big|_{\theta 1}\cdot\delta_i\delta_j\delta_k\delta_\ell\Big] \\ &+ \frac{1}{3!}\sum_{i,\ell,m,n}\frac{\partial^+h_p(t)}{\partial z_i}\Big|_E E\Big[\frac{\partial^{+3}h_q(t)}{\partial z_\ell\partial z_m\partial z_n}\Big|_{\theta 2}\cdot\delta_i\delta_\ell\delta_m\delta_n\Big] \\ &+ \frac{1}{12}\sum_{i,j,k,\ell,m}\frac{\partial^{+2}h_q(t)}{\partial z_\ell\partial z_m}\Big|_E E\Big[\frac{\partial^{+3}h_p(t)}{\partial z_i\partial z_j\partial z_k}\Big|_{\theta 1} \\ &\quad\cdot\delta_i\delta_j\delta_k(\delta_\ell\delta_m - Cov[z_\ell,z_m])\Big] \\ &+ \frac{1}{12}\sum_{i,j,\ell,m,n}\frac{\partial^{+2}h_p(t)}{\partial z_i\partial z_j}\Big|_E E\Big[\frac{\partial^{+3}h_q(t)}{\partial z_\ell\partial z_m\partial z_n}\Big|_{\theta 2} \\ &\quad\cdot\delta_\ell\delta_m\delta_n(\delta_i\delta_j - Cov[z_i,z_j])\Big] \\ &+ \frac{1}{36}\sum_{i,j,k,\ell,m,n}\Big(E\Big[\frac{\partial^{+3}h_p(t)}{\partial z_i\partial z_j\partial z_k}\Big|_{\theta 1}\frac{\partial^{+3}h_q(t)}{\partial z_\ell\partial z_m\partial z_n}\Big|_{\theta 2} \\ &\quad\cdot\delta_i\delta_j\delta_k\delta_\ell\delta_m\delta_n\Big] - E\Big[\frac{\partial^{+3}h_p(t)}{\partial z_i\partial z_j\partial z_k}\Big|_{\theta 1}\delta_i\delta_j\delta_k\Big] \\ &\quad\cdot E\Big[\frac{\partial^{+3}h_q(t)}{\partial z_\ell\partial z_m\partial z_n}\Big|_{\theta 2}\delta_\ell\delta_m\delta_n\Big]\Big), \qquad (12) \\ &\qquad (0<\theta_1<1, 0<\theta_2<1). \end{aligned}$$

The error bound of e_c can be derived similarly to the case of e_m, and for simplicity, the equation is omitted here.

For the case that $z_1, z_2, \cdots, z_\ell$ are white Gaussian, covariance and the error bound can be calculated as follows

$$\begin{aligned} &Cov[y_p(t),y_q(t)] \\ &= \sum_i\frac{\partial^+h_p(t)}{\partial z_i}\Big|_E\frac{\partial^+h_q(t)}{\partial z_i}\Big|_E\sigma_i^2 + e_c, \qquad (13) \\ &e_c \leq \frac{1}{2}\sum_i\left|\frac{\partial^{+2}h_p(t)}{\partial z_i^2}\Big|_E\frac{\partial^{+2}h_q(t)}{\partial z_i^2}\Big|_E\right|\sigma_i^4 \\ &\quad + \frac{1}{2}\sum_i\left|\frac{\partial^{+3}h_p(t)}{\partial z_i^3}\Big|_{max}\left|\frac{\partial^+h_q(t)}{\partial z_i}\Big|_E\right|\sigma_i^4 \\ &\quad + \frac{1}{2}\sum_i\left|\frac{\partial^+h_p(t)}{\partial z_i}\Big|_E\right|\left|\frac{\partial^{+3}h_q(t)}{\partial z_i^3}\Big|_{max}\sigma_i^4 \\ &\quad + \frac{5}{12}\sum_i\left|\frac{\partial^{+3}h_p(t)}{\partial z_i^3}\Big|_{max}\left|\frac{\partial^{+3}h_q(t)}{\partial z_i^3}\Big|_{max}\sigma_i^6. \qquad (14) \end{aligned}$$

3.3 Discussions

From equations (6) and (12) we can see that both of the error of mean e_m and the error of covariance e_c are related to the derivatives of the node outputs with respect to the stochastic inputs $\frac{\partial^+ h_p}{\partial z_i}$, $\frac{\partial^{+2} h_p}{\partial z_i \partial z_j}$, $\frac{\partial^{+3} h_p}{\partial z_i \partial z_j \partial z_k}$ and also related to the central moments of the stochastic inputs $Cor[z_i, z_j]$, μ_3, μ_4, μ_5, μ_6. The smaller the above derivatives and the central moments are, the smaller errors of means and covariances will become. As usual, it is not known what values of θ, θ_1, θ_2 are taken, so the errors cannot be directly calculated from the (6) and (12). Instead, (7), (14) give error bound of the means and covariances. For simplicity, only calculation about error of mean is discussed below, however, the results of these discussion can analogized to the calculation of error of covariance.

In the first, absolute value of derivative $\left|\frac{\partial^{+3} h_p}{\partial z_i \partial z_j \partial z_k}|_\theta\right|$ is replaced by its maximum value of $\left|\frac{\partial^{+3} h_p}{\partial z_i \partial z_j \partial z_k}\right|_{max}$. Extending $\frac{\partial^{+3} h_p}{\partial z_i \partial z_j \partial z_k}$ by the differential operating chain rule, this is equal to sum of the products of node outputs' derivatives. The sum of the products is also less that the sum of the products of absolute maximum values of node outputs, so, the smaller each order derivative of node functions is, the smaller error of mean will be. For example, In the case of common layered neural networks, if each node activation function is with small differential value, and the weights between the nodes are also small, then the error bound of mean will become small. Of course, we cannot set the node functions and the weights for a leaning network only by considering the accuracy of mean and covariance calculation. But different choice of node function and weight will lead to different learning results when the mean and covariance are a term of the criteria. It is easy to understand that choice of node function which has small differential value and weights with small value will lead to better learning results.

Secondly, from (7) we can see that e_m is related to the third moment of stochastic input μ_3 . The smaller μ_3 is, the smaller e_m will be. μ_3 is a quantity characterizing the stochastic input. We cannot change the characteristics of input signal, but in most applications, the signal-noise ratio is often less 1.0, so, if we do normalization on the input signals, then the value of each central moment will be less then 1.0. It is favorable to improve the accuracy of mean and covariance calculation.

It is have to be mentioned that the error bound of (7) is very loose, and it is very difficult to get the central moments greater than 3rd moment for general stochastic signals. But, the results obtained from (7) are very important in the sense of guidance of setting network node functions and weights. In addition, for the case of Gaussian white noise, calculation equation of mean (10) is exact one.

4 Conclusions

PrULN is a leaning network for dealing with stochastic signals. In PrULN scheme, some square errors of statistical values of node outputs are contained between actual values and desired values. Therefore, through the training, the statistical values of node outputs will approach desired one. Generally, different choice of node functions and initial weights, will lead to different learning result. It is necessary to estimate the errors of the statistical value of node outputs when the networks are applied to real systems.

In this paper, an approach for calculation of statistical values of node outputs from the statistical values of stochastic inputs was described, and a method for estimating the errors of mean and covariance was proposed. Although it is difficult to directly apply the equations to calculate the bounds of errors, but in the case of Gaussian white noise, we obtained relatively simple equation for calculating the error bound. In addition, through analyzing the factors which effect on the errors, it is concluded that choice of node function with smaller differential value, choice of smaller initial weights and normalization of input signals are advantageous to lead to learning networks which has desired statistic features.

References

[1] R. Isermann, S. Ernst, and O. Nelles, "Identification with dynamic neural networks – Architecture, comparisons, applications", in *Proc. of the 11th IFAC Symposium On System Identification*, Vol.3, pp. 997–1022 (1997).

[2] K. Hirasawa, M. Ohbayashi, J. Murata, J. Hu, "Probabilistic Universal Learning Network Theory" (in Japanese)C Trans. IEE of Japan, Vol.118–C, No.2, 224–230 (1998).

[3] J. Hu, K. Hirasawa, C.Z. Jin and K. Matsuoka, "Control of Nonlinear Systems Based on a Probabilistic Learning Network", in *Proc. of the 14th World Congress of IFAC* Vol.C pp. 447–452 (1999).

[4] K. Hirasawa, J. Hu, J. Murata, C.Z. Jin, K. Yotsumoto and H. Katagiri "Identification of Nonlinear Dynamic Systems by Using Probabilistic Universal Learning Networks", in *Proc. of IJCNN'99* (1998).

[5] Werbos, Beyond Regression: New Tools for Prediction and Analysis in the Behavior Science. PhD thesis. Harvard University (1974).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

The Convergency of GA for Neural Network

Akihiko Uchibori* Masato Oka* Taiji Satoh** Kanya Tanaka**

*Department of Mechanical Engineering
Ube National College of Technology
Ube, Yamaguchi, 755-8555 Japan

**Electrical and Electronic Engineering
Yamaguchi University
Ube, Yamaguchi, 755-8611 Japan

Abstract

We consider the existence of an optimal solution and the convergency of a genetic algorithm (GA) when it is used for Neural Network. The existence of an optimal solution is assured by the compactness of the the set of Neural Network. The convergency of the algorithm is assured by the compactness and the modification of the algorithm.

1 Introduction

In recent years, optimizing methods using Genetic Algorithm (GA) have been studied. But, the convergency of this algorithm and the existence of an optimal solution have not been discussed sufficiently.

In this paper, we foucused on the property of the target, in order to discuss the convergency of this algorithm. If the GA is used for optimizing a Neural network, an infinite sequence of sets of production rules is generated. In order to discuss the convergency of this algorithm, the convergency of the infinite sequence of the network sets must be examined and the compactness of its sets becomes important.

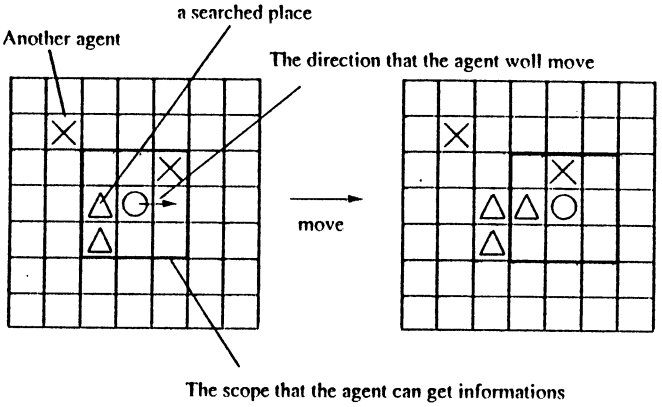
In this paper, we consider the problem of optimizing neural network using the GA. The convergency of this algorithm and the existence of an optimal solution are examined by using the compactness mentioned above.

2 Optimizing Neural Network with the GA

The process of optimizing neural network with the GA is explained, by using autonomous searching agents as an example. Here, 'Agents search for an object in a square area in cooperation with their own families'. The agents can only see around themselves, not among the entire area. In order to where they should be searching next, they should look around and consult with other agents near them.

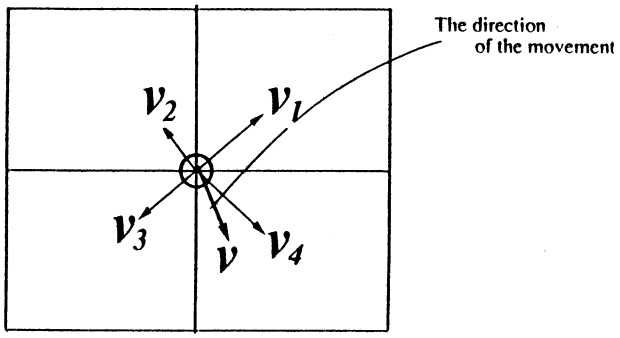
Dividing equally the area around an agent into n-directions, the moving distance of the agent is defined as $m_1, m_2, \ldots, m_n$. The moving distance should be decided by the following neural network $\mu \in F(\Delta, \gamma)$. $F(\Delta, \gamma)$ defined as follows:

$$\begin{aligned}
&\Delta, \gamma > 0, \\
&\psi(x) = \frac{1}{1+e^{-x}}, \\
&|\phi_\delta(x') - \phi_\delta(x)| \leq |\delta x' - \delta x| \\
&\quad = \delta |x' - x| \leq \Delta |x' - x|, \\
&F(\Delta, \gamma) \stackrel{\Delta}{=} \{\mu : x \in B_D \mapsto \mu(x) \in \boldsymbol{R}, \\
&|w|,\ |\theta| \leq \gamma,\ 0 < \delta < \Delta, \\
&|\phi_\delta(x) - \phi_\delta(x')| \leq \Delta |x - x'|, \}
\end{aligned} \tag{1}$$



(a) Getting informations of other objects and searched places

Fig. 1(a) Decision of the moving direction



(b) An agent gets the direction of the movement
by the composition of each vectors

Fig. 1(b) Distance moved towards the direction

The value m_k that shows the distance moved towards the direction k is obtained by neural network. There are n directions, therefore we obtain $m_1, \ldots, m_n$. We make n vectors from them as follows: (see Fig. 1(b)).

$$\begin{aligned} v_k &= (m_k \cos(\frac{2\pi}{n}k), \\ &\quad m_k \sin(\frac{2\pi}{n}k)) \\ &\quad k = 1, 2, \cdots, n \end{aligned} \tag{2}$$

By combining these vectors, we get the distance and direction of the moving agent.

We use the GA in order to optimize the neural network of the n pairs of solution above.

In this simulation, the neural networks μ_k are encoded to Fourier series approximations, and the A, a, b coefficients are treated as the information of the gene. By selecting, crossing, and mutating them, we optimize the algorithm.

$$\begin{aligned} \mu(\theta) &= \sum_m A_m \{a_m \cos(m\theta) + b_m \sin(m\theta)\} \\ &\quad (0 \leq \theta < 2\pi) \end{aligned} \tag{3}$$

The searching agent o is identified by a neural network, so we can express it as follows:

$$o = (\mu_1, \mu_2, \cdots, \mu_n) \in F(\Delta, \gamma)^n \tag{4}$$

The set of agents E consists of Q agents o which have the same genes and Fig. 2 shows the situation of its action.

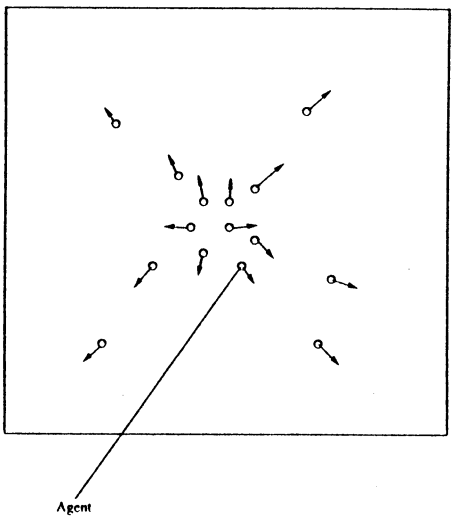


Fig. 2 Agent group

The number of agent sets E in the family of a generation G is constant N. (*i.e.* : $G = (E^1, \cdots, E^l, \cdots, E^N)$)

The total number of all agents U is as follows:

$$U = F(\Delta, \gamma)^n \tag{5}$$

Because the number of agents in a generation G is constant, the set of all generations Π is $\Pi = U^N$.

The fitness value of the agent set J_E is the total area of searched places.

The fitness function of generation $G = (E^1, \cdots, E^l)$ is as follows:

$$J(G) = \sum_{l=1}^{N} J_E^l \tag{6}$$

Simulations are carried out under the above conditions. The algorithm is as follows:

1. Decide a small value γ_0
   Create searching agent families as a first generation (G_0) which have chromosomes generated at random base.
   Obtain the fitness values $J_{E_0}^l$ of $E_0^l \in G_0$, $(l = 1, 2, \cdots, N)$ and $J(G_{i+1})$.
   Set $i = 0$.

2. Generate the $i+1$-th generation G_{i+1} from i-th generation G_i with the GA.

3. Applying the neural network
compute $J_{E_{i+1}}^{l}$ $(l = 1, 2, \cdots, n)$ and $J(G_{i+1})$.

4. If $J(G_i) + \delta_0 < J(G_{i+1})$, set $i = i + 1$, generate G_{i+1} from G_i then go to #3.
If not stop.

Fig. 3 shows the simulation result.

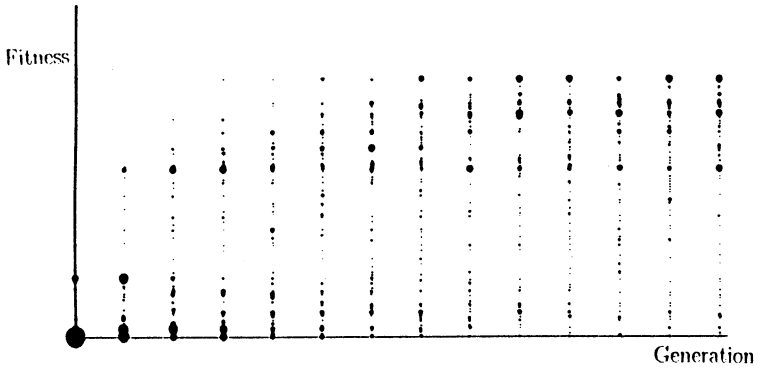


Fig. 3 Result of simulation

3 Convergency and Existence of a Solution

Fig. 4 shows that the fitness values J_E of agents E converges to some point. We could observe that agents that have quite different properties (genes) from each other can converge to some groups of characteristic.

Next, we consider the existence of the optimal solution and the convergency of the algorithm.

3.1 Existence of an optimal solution

The existence of maximum value of the fitness function

$$G \in \Pi \mapsto J(G) \tag{7}$$

of generation G is discussed in below.

$F(\Delta, \gamma)$ is a subset of the entire continuous functions set $C(B_D, \boldsymbol{R})$ on $B_D \subseteq \boldsymbol{R}$. In that case, $F(\Delta, \gamma)$ is compact with respect to uniform topology. Therefore, all unions of conclusion parts which have been cut and obtained by the process of applying production rules belong to this set.

The direct product of compact space $F(\Delta, \gamma)$

$$\Pi = U^N = (F(\Delta, \gamma)^n)^N = F(\Delta, \gamma)^{nN}$$

is also compact. $J(G)$ is continuous with respect to G. It is well known that continuous functions in compact space have a maximum (minimum) value. Therefore, $J(G)$ have maximum values in Π. This means that a generation G which makes $J(G)$ a maximum exists in the set Π of all generations.

3.2 Convergency

Using the E. Polak's theorem[2], we consider the convergency of the algorithm.

The GA in the simulation generates a new $G_{i+1} \in \Pi$ from $G_i \in \Pi$. This is shown by the following function

$$a : \Pi \to \Pi \tag{8}$$

That is,

$$G_{i+1} = a(G_i) \tag{9}$$

Here, a is

$$a(G) = \begin{cases} G' & (G' \in S(G)) \\ G & (S(G) = \phi) \end{cases} \tag{10}$$

$$S(G) = \{G' \mid G' \in \Pi; J(G') > J(G)\} \tag{11}$$

(10) and (11) are the additional modification of genetic algorithm in order to assure the convergency.

Let generation G as the local maximum value in $J(G)$ be a feasible solution. According to E. Polak[2], If a and $J(G)$ satisfy the following conditions,

(i) $J_G$is continuous.

(ii) $(\forall G \in \Pi$; non-feasible solutions),
$\exists \varepsilon(G)$, $\exists \delta(G) > 0$,
$\forall G' \in B(G, \varepsilon(G))$,
$J(a(G')) - J(G') \geq \delta(G) > 0$
$B(G, \varepsilon(G)) = \{G' \in \Pi \mid d(G', G) < \varepsilon(G)\}$,
$d(G', G) =$

$$\sum_{l=1}^{N} \sum_{k=1}^{n} \sum_{r=1}^{2} (\| \mu_{if_k^r}^{l} - \mu_{if_k^r}^{l} \|_\infty + \| \mu_{then_k^r}^{l} - \mu'^{l}_{then_k^r} \|_\infty) \tag{12}$$

then,

- $\{G_i\}$ generated by the algorithm is a finite sequence and the final element of the sequence is a feasible solution,
- or it is an infinite sequence and any accumulation points is a feasible solution.

$J(G)$ is continuous of (i) above. If a is a continuous operator with respect to G and $\varepsilon(G)$ is sufficiently small, condition (ii) is satisfied. Also, Π is compact and any infinite sequence $\{G_i\}$ has accumulation points. This means that $\{G_i\}$ eventually converges to a feasible solution, but it is not known how many i of G_i are needed for convergence.

The above discussion is a consideration of the searching agents case. We believe that this discussion applies not only upon the example here, but also upon the general problems. The key points of this paper are about the few restrictions of neural networks that guarantee the existence of an optimal solution, and a revision of the traditional GA, that guarantees the convergency of the algorithm.

4 Conclusion

In optimizing neural network with the GA, we were able to discuss the convergency and the existence of the optimal solution of the algorithm by applying it to a simple retrieval (search) problem. To assure the convergency, the GA was reformed. Also, to assure the existence of the optimal solution, neural networks were given the few restrictions. But, in most cases, they did not have to be considered. This algorithm can be applied widely upon other problems that optimize neural network with GA, not only the example in this paper.

These methods are not strong enough to assure that every GA will converge to an optimal solution. But, we hope this work will serve as a first step in building a theoretical base for GAs.

References

[1] C.L.Karr, "Design of an Adaptive Fuzzy Logic Controller Using a Genetic Algorithm", Proc. of 4th Int. Conf. on Genetic Algorithm, pp.450-457, 1991.

[2] E. Polak, Computational Methods in Optimization, New York, Academic, 1971.

[3] A. Uchibori, H. Yamazaki, Y. Shidama and H. Yamaura, "An Optimizing Fuzzy Logic with Genetic Algorithm", proc. ICEC'96 1996 IEEE 3rd International Conference on Evolutionary Computation, Nagoya, Japan, pp.239-242 May, 1996

Appendix

E. Polak's Theorem

We cite the abridged proof of E. Polak's theorem in Sec. 3.

When

$$J(a(G)) \leq J(G) \tag{13}$$

is satisfied, it is obvious by condition 2 that G is a feasible solution.

If $\{G_i\} = \{G_0, G_1, \cdots, G_k, G_{k+1}\}$ are finite sequences, then G_k is a feasible solution by [Step 3] and expression (13). Let $\{G_i\}$ be infinite sequences that include a subsequence that converges to G'. Let $G_i \to G' \quad i \in K \subset \{1, 2, \cdots\}$. Here, if G' is not a feasible solution, $\varepsilon' > 0$, $\delta' < 0$, $\forall i \geq k, i \in K$ exist and it becomes as follows:

$$\| G_i - G' \| \leq \varepsilon' J(G_{i+1})) - J(G_i) \geq \delta(G) \tag{14}$$

Therefore, all G_i and G_{i+1} sitting side by side in a subsequence become, for $i \geq k (i, (i+j) \in K)$,

$$\begin{aligned} & J(G_{i+j}) - J(G_i) \\ & = [J(G_{i+j}) - J(G_{i+j-1})] \\ & \quad + [J(G_{i+j-1}) - J(G_{i+j-2})] + \cdots \\ & \quad + [J(G_{i+1}) - J(G_i)] \\ & > J(G_{i+1}) - J(G_i) \geq \delta'. \end{aligned} \tag{15}$$

Here, if $J(G_i)$ is a monotone increasing sequence in $i \in K$, $J(\cdot)$ are continuous with respect to G'. Therefore, it must converge. But, this is inconsistent with expression (15) which shows that the sequence $J(G_i)$ is not a Cauchy sequence in $i \in K$.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Independent Component Analysis Using Time Delayed Sampling

Michifumi Yoshioka Sigeru Omatu
College of Engineering
Osaka Prefecture University
Sakai, Osaka 599-8531, Japan

Abstract

In recent years, growing multimedia systems require more efficient signal separation methods to preserve the quality of voice or music recording under a noisy environment. Some of signal separation methods are based on minimizing the dependent measure among input signals to separate a noise component since a noise component is usually independent on the other signals. Under such circumstances, we have developed a new method to separate independent signal components which directly minimizes the Kullback-Leibler divergence by a genetic algorithm(GA). In this paper, we have proposed an improved method using differential information. As the result of the simulation, the separated signals are clearly separated to original signals.

1 Introduction

In this paper, we have proposed a new method of blind signal separation using GA. The blind signal separation is the problem to obtain independent components from mixed input signals without knowing original signals and mixing matrix, where input signals are mixture of original signals which are mutually independent. Many blind signal separation methods have been proposed using neural networks. However, The performance of these methods are affected by the activation function of neurons and the distribution of input signals. The proposed method directly minimizes the Kullback-Leibler divergence among input signals by GA. Therefore, the selection of activation functions which fit the distributions of input signals is useless.

2 Overview of the proposed signal separation method

The problem which is considered in this paper is formulated as follwos. Input signals $\boldsymbol{x}_i$ are mixture of original signals $\boldsymbol{s}_i$ which are mutually independent.

$$\boldsymbol{x}_i = \boldsymbol{R}\boldsymbol{s}_i \tag{1}$$

where,

$$\boldsymbol{x}_i = \begin{pmatrix} x_i^1 \\ \vdots \\ x_i^n \end{pmatrix}, \boldsymbol{s}_i = \begin{pmatrix} s_i^1 \\ \vdots \\ s_i^m \end{pmatrix}$$

When $\boldsymbol{R}$ is unknown n x m matrix, the problem is to find the separation matrix $\boldsymbol{W}$ and the estimation of original signals $\hat{\boldsymbol{s}}$.

$$\hat{\boldsymbol{s}}_i = \boldsymbol{W}\boldsymbol{x}_i \tag{2}$$

In this paper, we have emploied Kullback-Leibler divergence to evaluate independence among separated signals $\hat{\boldsymbol{s}}$. Kullback-Leibler divergence is defined as follows:

$$\int p(\hat{\boldsymbol{s}}) log \frac{p(\hat{\boldsymbol{s}})}{\prod p_i(\hat{s}^i)} d\hat{\boldsymbol{s}} \tag{3}$$

The function $p(\hat{\boldsymbol{s}})$ and $p_i(\hat{s}^i)$ are the probability density function and the marginal probability density function of $\hat{\boldsymbol{s}}$ and $\hat{s}^i$ respectively. The equation 3 is rewritten by the marginal entropy as follows:

$$-H(\hat{\boldsymbol{s}}) + \sum_{i=1}^{n} H(\hat{s}^i) \tag{4}$$

where

$$H(\hat{\boldsymbol{s}}) = -\int p(\hat{\boldsymbol{s}}) log p(\hat{\boldsymbol{s}}) d\hat{\boldsymbol{s}} \tag{5}$$

$$H(\hat{s}^i) = -\int p_i(\hat{s}^i) log p_i(\hat{s}^i) d\hat{s}^i \tag{6}$$

In order to calculate the above integral from input signals $\hat{\boldsymbol{s}}$, the calculation is performed according to the following steps.

1. Separate the integral area into N_d areas for each input signal component between the minimum value $\hat{s}^i_{min}$ and the maximum value $\hat{s}^i_{max}$, and $N_{dt} = N_d^{N_c}$ areas for whole input signals where N_c is the number of signal components.

2. Classify the input signals $\hat{s}^i$ into areas divided in 1, and count the classified signals as $C_{dt}(n_1, n_2, ..., n_{N_c})$. That is

$$N_s = \sum^{N_{dt}} C_{dt}(n_1, n_2, ..., n_{N_c}) \tag{7}$$

where N_s is the total number of sampled signals.

3. In order to obtain marginal distribution, Accumulate the result of step 2 for each component.

$$C_d(n_j) = \sum^{N_{dt}/N_d} C_{dt}(n_1, n_2, ..., n_j, ..., n_{N_c}) \tag{8}$$

4. From the result of step 2, 3, approximate the probability density function $p(\hat{s})$ and the marginal probability density function $p_j(\hat{s}^j)$ using $C_{dt}(n_1, n_2, ..., n_{N_c})$ and $C_d(n_j)$ as follows:

$$p(\hat{s}) = \frac{C_{dt}(n_1, n_2, ..., n_{N_c})}{N_s} N_{dt} \tag{9}$$

$$p_j(\hat{s}^j) = \frac{C_d(n_j)}{N_s} N_d \tag{10}$$

where,

$$\frac{1}{N_d}(n_j - 1) \leq \frac{\hat{s}^j - \hat{s}^j_{min}}{\hat{s}^j_{max} - \hat{s}^j_{min}} < \frac{1}{N_d} n_j \tag{11}$$

The specification of GA which is used to find the separation matrix $\boldsymbol{W}$ is as follows.

1. Make genes which are elements of the separation matrix.

2. Sort genes according to the independence evaluated by the equation (3).

3. Select genes using the roulette selectin.

4. Perform mutation and crossover.

The GA parameters which are used in all simulation are shown in Table 2.

3 Simulation

In order to evaluate the proposed method, we have simulated the method on computer using the original signals defined by the equation (12) and illustrated in Fig. 1.

Table 1: GA parameters

selection operator	roulette selection
crossover operator	one point crossover
mutation	0.1
genes	50
generations	50

$$\begin{aligned} \boldsymbol{s}_i &= \begin{pmatrix} s_i^1 \\ s_i^2 \\ s_i^3 \end{pmatrix} \\ &= \begin{pmatrix} sin(5t + cos(4t)) \\ \text{white noise} \\ sign[sin(5t + cos(8t))] \end{pmatrix} \end{aligned} \tag{12}$$

where $t = i/50$

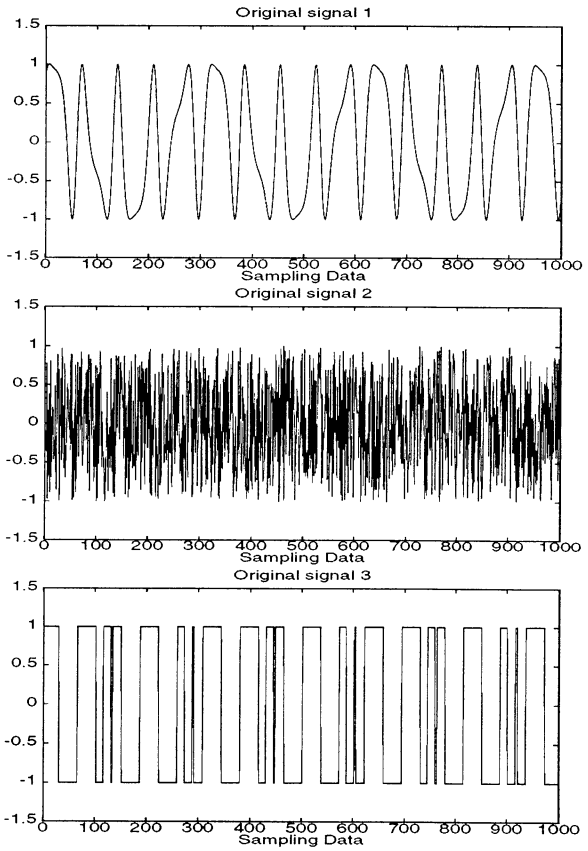


Figure 1: Original signals.

Those original signals $\boldsymbol{s}$ are mixed to synthesis the

mixed signals $\boldsymbol{x}$ using the following mixing matrix $\boldsymbol{R}$ whose elements are randomly generated in [-1,1].

$$\boldsymbol{R} = \begin{pmatrix} 0.400649 & 0.823427 & -0.000604 \\ 0.731718 & -0.784769 & -0.786887 \\ 0.811587 & 0.472167 & 0.876216 \end{pmatrix} \quad (13)$$

The mixed signals $\boldsymbol{x}$ are illustrated in Fig. 2.

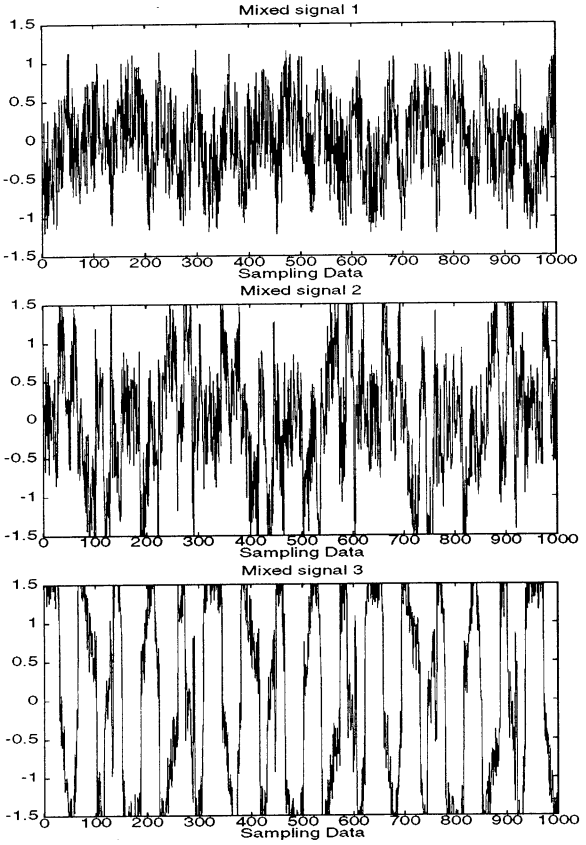


Figure 2: Mixed signals.

In this simulation, the used input signals are 1000 points sampled from the mixed signals $\boldsymbol{x}$.

As the result of the simulation, the separation matrix $\boldsymbol{W}$ is

$$\boldsymbol{W} = \begin{pmatrix} -0.493 & -0.910 & -0.764 \\ 0.832 & 0.088 & 0.244 \\ 0.678 & 0.630 & -0.212 \end{pmatrix} \quad (14)$$

And, the separated signals $\hat{\boldsymbol{s}}$ which are calculated by the separation matrix $\boldsymbol{W}$ are illustrated in Fig. 3.

Finally, the product of matrix $\boldsymbol{W}$ and $\boldsymbol{R}$ which shows the relation of the original and the separated signals is

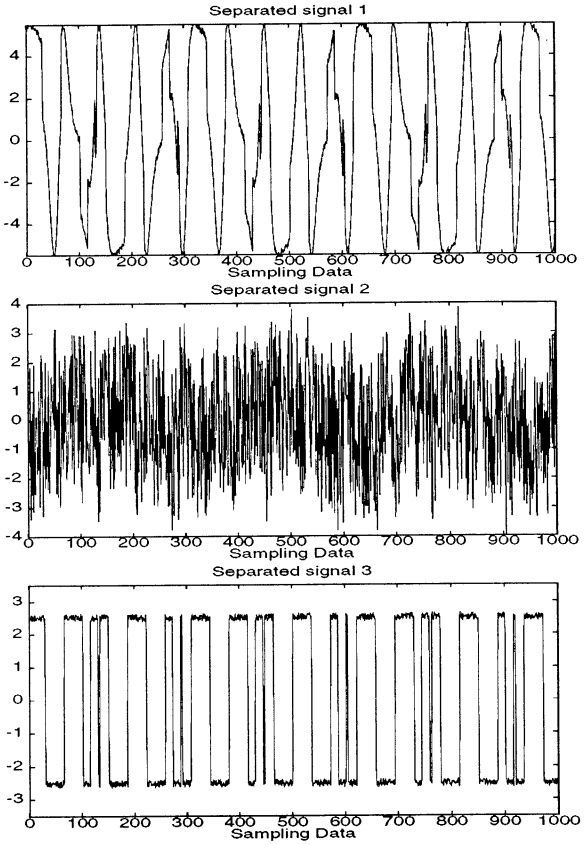


Figure 3: Separated signals 1.

$$\boldsymbol{W}\boldsymbol{R} = \begin{pmatrix} -1.088 & -0.053 & 0.047 \\ -0.071 & 0.731 & 0.144 \\ 0.017 & -0.036 & -0.682 \end{pmatrix} \quad (15)$$

4 Improved method

The proposed method shows good performance as the above simulation. However, the method requires more mixed signals than separated signals, because the separated signals are estimatied using linear combination of mixed signals at the same time. We have imporved the method about this point.

To achive the above improvement, we have just adopted the differential informaition of time series input data $\dot{\boldsymbol{x}}_i$.

In the improved method(method 2), the estimations of original signals are expressed by the following equation,

$$\hat{\boldsymbol{s}}_i = \boldsymbol{W} \begin{pmatrix} \boldsymbol{x}_i \\ \dot{\boldsymbol{x}}_i \end{pmatrix} \tag{16}$$

while original proposed method(method 1), the estimations are expressed by the equation (2).

For example, in the case of two mixed signals and three separated siganals, the equation (16) is writen as follows,

$$\hat{\boldsymbol{s}}_i = \begin{pmatrix} \hat{s}_i^1 \\ \hat{s}_i^2 \\ \hat{s}_i^3 \end{pmatrix} \cong \boldsymbol{W} \begin{pmatrix} x_i^1 \\ x_i^2 \\ x_{i+1}^1 - x_i^1 \\ x_{i+1}^2 - x_i^2 \end{pmatrix} \tag{17}$$

In the above case, the separation matrix $\boldsymbol{W}$ have 4 columns and 3 rows.

To evaluate new method, we have simulated those methods. The same parameters are used in both methods without the number of mixed signals as input. The mixing matrix $\boldsymbol{R}_2$ in the method 2 is

$$\boldsymbol{R}_2 = \begin{pmatrix} 0.731718 & -0.784769 & -0.786887 \\ 0.811587 & 0.472167 & 0.876216 \end{pmatrix} \tag{18}$$

The result of simulation comparing method 1 and method 2 is shown in Table 2. The simulation is performed in 50 times and the indicator which is used in Table 2 is the average of squared error from original signals. The result of simulation shows good performance in the method 1,2.

Table 2: Separation Performance(methods 1, 2).

	s^1	s^2	s^3
Method 1	0.059	0.003	0.007
Method 2	0.149	0.841	0.047

5 Conclusions

In this paper, we have developed the independent component separation filter using Kullback-Leibler divergence and GA, and improved the method by differential informaition. As the results of simulations illustrated in Fig. 1-4, it is clear that the independent component separation ability of the proposed method is promised but preliminary. As a future problem, we will apply this method to various signal processings as a preprocessing filter.

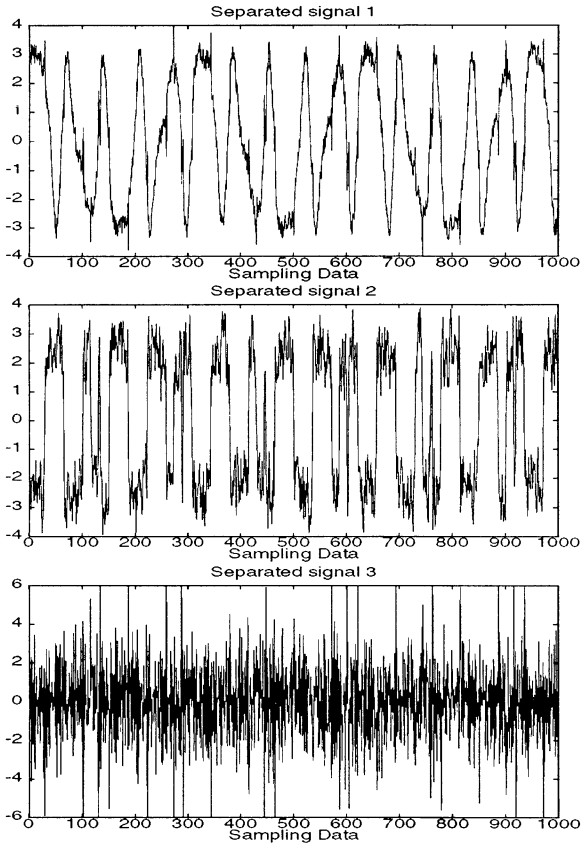


Figure 4: Separated signals 2.

References

[1] S.Amari, A.Cichocki, H.H.Yang, " A New Learning Algorithm for Blind Signal Separation", *Advances in Neural Information Processing Systems*, Vol.8, pp.757-763, 1995.

[2] E. Oja., "The Nonlinear PCA Learning Rule and Signal Separation - Mathematical Analysis", *Technical Report A26*, Helsinki University of Technology, Lab. of Computer and Information Science, 1995.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Design and Experimental Evaluation of a Neural-Net Based Multivariable PID Controller

Toru Yamamoto
Dept. of Technology Education
Hiroshima University
Hiroshima, 739-8524, *Japan*
toyamam@sed.hiroshima-u.ac.jp

Toshitaka Oki
Dept. of Communication Eng.
Okayama Prefectural University
Okayama, 719-1197, *Japan*
oki@c.oka-pu.ac.jp

Sirish L. Shah
Dept. of Chemical and Materials Eng.
University of Alberta
Alberta, T6G 2G6, *Canada*
sirish.shah@ualberta.ca

Abstract

In this paper, a design scheme of neural-net based controllers with a PID structure is proposed for the control of such multivariable systems. The proposed controller consists of a pre-compensator designed by a static gain matrix which compensates for the low-frequency interaction, and PID controllers placed diagonally whose gains are tuned by a neural network.

Key Word : neural network, PID control, multivariable system, process control, decoupling

1 Introduction

PID[1]-[2] control algorithms still continue to be widely used for most industrial control systems, particularly in the chemical processes. This is mainly because PID controllers have simple control structures, and are simple to maintain and tune. Furthermore, it is well known that process systems have multi-inputs and multi-outputs, and strong mutual interactions between inputs and outputs. For multivariable systems with interactions, authors have proposed a design scheme of multiloop PID controllers[3] from the viewpoint of self-tuning schemes, and has been experimentally evaluated.

In this paper, a new design method of neural-net based PID controller for multivariable systems is proposed. The proposed controller consists of a set of diagonal PID controllers, one par loop and a static matrix pre-compensator matrix which compensates for the law frequency interactions. The static pre-compensator renders many practical multivariate systems diagonally dominant. PID controllers can then designed for each loop of the pre-conpensated system independently. Therefore, we have only p-PID controllers to design for a p-input and p-output multivariable system. The PID gains of this proposed controller are composed of the fixed PID gains based on *a priori* information about the system and incremental or fine-tuned PID gain updates by the neural network. The main purpose in this study is to present a multiloop PID control scheme so that the interaction may be removed by a neural network.

2 Multivariable Neural-Net Based PID Controller

2.1 Mathematical model

Let z^{-1} be the backward shift operator, then the following discrete-time p-input and p-output multivariable description of ARX system describes the process:

$$\boldsymbol{A}(z^{-1})\boldsymbol{y}(t) = D\boldsymbol{B}(z^{-1})\boldsymbol{u}(t-1) \tag{1}$$

where $\boldsymbol{u}(t)$ and $\boldsymbol{y}(t)$ are the input and output vector with p-elements, *i.e.*,

$$\left.\begin{aligned}\boldsymbol{u}(t) &= [u_1(t), u_2(t), \ldots, u_p(t)]^T \\ \boldsymbol{y}(t) &= [y_1(t), y_2(t), \ldots, y_p(t)]^T.\end{aligned}\right\} \tag{2}$$

Furthermore, $\boldsymbol{A}(z^{-1})$ is a diagonal polynomial matrix given by

$$\boldsymbol{A}(z^{-1}) = \mathrm{diag}\{A_1(z^{-1}), A_2(z^{-1}), \cdots, A_p(z^{-1})\} \tag{3}$$

where each $A_i(z^{-1})$ is at most a second-order scalar polynomial of the form:

$$A_i(z^{-1}) = 1 + a_{i,1}z^{-1} + a_{i,2}z^{-2} \quad (i = 1, 2, \cdots, p) \tag{4}$$

$\boldsymbol{B}(z^{-1})$ is the following polynomial matrix with full elements:

$$\begin{aligned}&\boldsymbol{B}(z^{-1}) \\ &= \begin{bmatrix} B_{11}(z^{-1}) & B_{12}(z^{-1}) & \ldots & B_{1p}(z^{-1}) \\ B_{21}(z^{-1}) & B_{22}(z^{-1}) & \ldots & B_{2p}(z^{-1}) \\ \vdots & \vdots & \ddots & \vdots \\ B_{p1}(z^{-1}) & B_{p2}(z^{-1}) & \ldots & B_{pp}(z^{-1}) \end{bmatrix} \\ &= B_0 + B_1 z^{-1} + \ldots + B_m z^{-m},\end{aligned} \tag{5}$$

where the coefficient matrix B_k is defined as

$$B_k = \begin{bmatrix} b_{k,1,1} & b_{k,1,2} & \ldots & b_{k,1,p} \\ b_{k,2,1} & b_{k,2,2} & \ldots & b_{k,2,p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{k,p,1} & b_{k,p,2} & \ldots & b_{k,p,p} \end{bmatrix}. \tag{6}$$

D is the time-delay matrix of the form:

$$D = \mathrm{diag}\{z^{-k_1}, z^{-k_2}, \ldots, z^{-k_p}\}. \quad (7)$$

k_i denotes the time-delay to the ith output $y_i(t)$ from every input signal. That is, it is assumed that the time-delay to each output from every input is same. Therefore, all entries of B_0 in (5) are non-zero.

2.2 Pre-compensator

In designing controllers for multivariable systems, it is important to first remove or compensate interactions. The simplest pre-compensator that can be designed as a static gain matrix, H is with the form[4]:

$$H := \boldsymbol{B}^{-1}(1)\boldsymbol{A}(1) = \begin{bmatrix} h_{11} & h_{12} & \ldots & h_{1p} \\ h_{21} & h_{22} & \ldots & h_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ h_{p1} & h_{p2} & \ldots & h_{pp} \end{bmatrix}. \quad (8)$$

Note that such a pre-compensator essentially looks after the low frequency interaction.

Then, the augmented system constructed by (1) and (8) can be described as

$$\boldsymbol{A}(z^{-1})\boldsymbol{y}(t) = D\boldsymbol{B}(z^{-1})H\boldsymbol{v}(t-1) \quad (9)$$

where $\boldsymbol{v}(t)$ denotes the input signal vector to the augmented system. By regarding the augmented system (9) as the approximately or almost decoupled system, the following model can be obtained for each diagonal element:

$$A_i(z^{-1})y_i(t) = z^{-k_i}\bar{B}_i(z^{-1})v_i(t-1) \quad (i = 1, 2, \ldots, p) \quad (10)$$

where

$$\bar{B}_i(z^{-1}) := \sum_{k=1}^{p} B_{ik}(z^{-1})h_{ki} = \bar{b}_{i,0} + \bar{b}_{i,1}z^{-1} + \cdots + \bar{b}_{i,m}z^{-m} \quad (11)$$

and $A_i(z^{-1})$ is given by (4).

2.3 PID control system

Next, we design PID controller for the augmented system given by (10). The digital velocity-type PID control law to be considered in this paper is described as

$$\Delta v_i(t) = K_{Ii}e_i(t) - \left(K_{Pi}\Delta + K_{Di}\Delta^2\right)y_i(t) \quad (12)$$

where $e_i(t)$ denotes the control error signal given by

$$e_i(t) = w_i(t) - y_i(t), \quad (13)$$

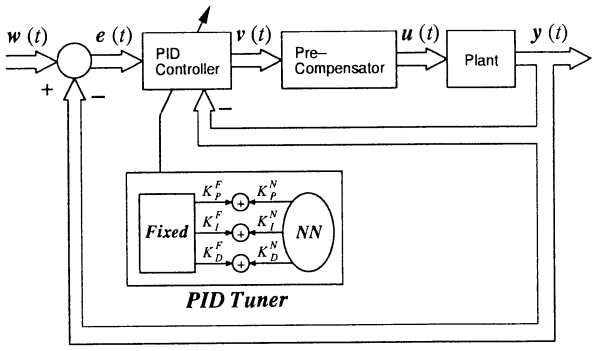


Figure 1: Schematic diagram of the multivariable neural-net based PID control system.

and, $w_i(t)$ denotes the ith reference output. Furthermore, Δ is the differential operator defined by $\Delta := 1 - z^{-1}$. And K_{Pi},K_{Ii} and K_{Di} are the proportional, the integral and the differential gains, respectively, and defined by

$$\begin{cases} K_{Pi} = K_{Pi}^F + K_{Pi}^N \\ K_{Ii} = K_{Ii}^F + K_{Ii}^N \\ K_{Di} = K_{Di}^F + K_{Di}^N. \end{cases} \quad (14)$$

In (14), the subscripts 'F' denotes the fixed PID gains, and 'N' denotes the PID gains tuned by the neural network. The fixed PID gains are calculated based on *a priori* information about the system.

The schematic diagram of the multivariable neural-net based PID control system to be designed is shown in Figure 1. As mentioned above, interactions can not be compensated strictly by using only the pre-compensator H. Then, by using a neural network, K_{Pi}^N,K_{Ii}^N and K_{Di}^N are tuned so that interactions may be removed. Therefore, note that PID gains are not fixed, and change for each reference signal.

2.4 Neural network

In this section, the structure and the training rules of the neural network[5] used in this work are discussed. The neural network is constructed with three layers as shown in Fig.2. Here, $\mathcal{I}_j(t)$,$\mathcal{H}_k(t)$ and $\mathcal{O}_l^i(t)$ are the output signal of the jth unit in the input layer, the kth unit in the hidden layer and the lth unit in the output layer for ith control input, and $\square$ denotes the threshold unit. And q denotes the number of units in the input layer, r denotes the number of units in the hidden layer for each control input. Furthermore, both $V_{jk}(t)$ and $W_{kl}^i(t)$ denote the connection weights between succeeding layers. Here, $\mathcal{H}_k(t)$ and $\mathcal{O}_l^i(t)$ are respectively calculated by

$$\mathcal{H}_k(t) = f\Big(\sum_{j=0}^{q-1} V_{jk}(t)\cdot\mathcal{I}_j(t)\Big) \quad (15)$$

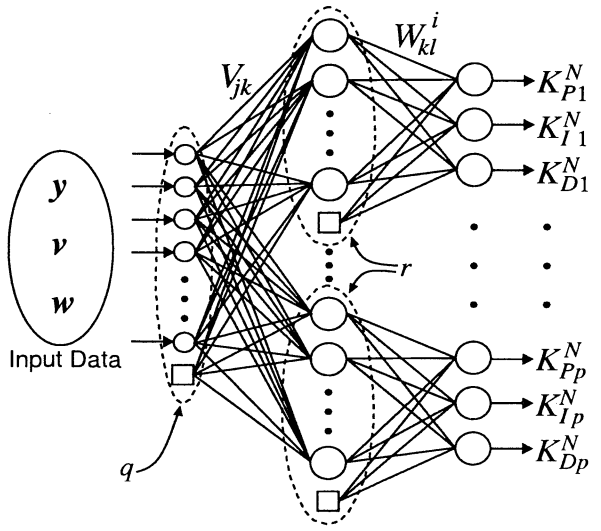


Figure 2: Schematic figure of the neural network.

$$\mathcal{O}_l^i(t) = \beta \cdot f\Big(\sum_{k=r(i-1)}^{ri-1} W_{kl}^i(t) \cdot \mathcal{H}_k(t)\Big) \qquad (16)$$
$$(\beta > 0)$$

where $\mathcal{I}_0(t)$ and $\mathcal{H}_{r(i-1)}(t)$ denote the threshold unit. And $f(\cdot)$ denotes the sigmoidal function given by

$$f(x) := \frac{1}{1+e^{-\alpha x}} - \frac{1}{2} \qquad (\alpha > 0). \qquad (17)$$

Then, note that there are relations between the output layer and $K_{P\,i}^N, K_{I\,i}^N, K_{D\,i}^N$ as follows:

$$\left\{ \begin{array}{l} K_{P\,i}^N = \mathcal{O}_1^i(t) \\ K_{I\,i}^N = \mathcal{O}_2^i(t) \\ K_{D\,i}^N = \mathcal{O}_3^i(t). \end{array} \right. \qquad (18)$$

The weights $V_{jk}(t)$ and $W_{kl}^i(t)$ included in the neural network, are updated so that following criterion $E(t)$ may be decreased:

$$E(t) := \frac{1}{2}\sum_{i=1}^{p}\{\varepsilon_i(t)\}^2. \qquad (19)$$

$$\varepsilon_i(t) = y_{mi}(t) - y_i(t), \qquad (20)$$

where $y_{mi}(t)$ denotes the ith reference model output.

The updated rules of the weights are derived based on backpropagation method. Then, these rules can be obtained as follows:

$$W_{kl}^i(t) = W_{kl}^i(t-k_i-1) - \eta_1 \delta_{1,l}^i(t)\mathcal{H}_k(t-k_i-1) \quad (21)$$

$$V_{jk}(t) = V_{jk}(t-k_i-1) - \eta_2 \delta_{2,k}(t)\mathcal{I}_j(t-k_i-1) \quad (22)$$

where $x_l^i(t)$ is defined as

$$x_l^i(t) = \left\{ \begin{array}{ll} -\Delta y_i(t) & (l=1) \\ e_i(t) & (l=2) \\ -\Delta^2 y_i(t) & (l=3), \end{array} \right. \qquad (23)$$

η_1 and η_2 are the positive constants, and $\delta_{1,l}^i(t)$ and $\delta_{2,k}(t)$ are respectively defined as

$$\begin{aligned} \delta_{1,l}^i(t) = & -\frac{\alpha}{\beta}\sum_{n=1}^{p}\Big(b_{0,i,n}h_{ni}\Big)\varepsilon_i(t)x_l^i(t-k_i-1) \\ & \cdot\left\{\frac{\beta}{2} + \mathcal{O}_l^i(t-k_i-1)\right\} \\ & \cdot\left\{\frac{\beta}{2} - \mathcal{O}_l^i(t-k_i-1)\right\} \end{aligned} \qquad (24)$$

$$\begin{aligned} \delta_{2,k}(t) = & \alpha\left\{\frac{1}{2} + \mathcal{H}_k(t-k_i-1)\right\} \\ & \cdot\left\{\frac{1}{2} - \mathcal{H}_k(t-k_i-1)\right\} \\ & \cdot\sum_{l=1}^{3}\delta_{1,l}^i W_{kl}^i(t-k_i-1). \end{aligned} \qquad (25)$$

The connection weights $V_{jk}(t)$ and $W_{kl}^i(t)$ of the neural network are calculated based on the update rule (22) and (21), respectively.

Then, the multivariable neural-net based PID control system can be designed, which follows the reference model output $\boldsymbol{y}_m$.

3 Simulation Example

In this section, a numerical simulation example is discussed in order to investigate the behavior of the newly proposed control scheme. We consider the following nonlinear multivariable system with 2-inputs and 2-outputs:

$$\left.\begin{array}{rl} y_1(t) = & \dfrac{y_1(t-1)}{1+y_1^2(t-1)} + 0.2y_2(t-1) \\ & +0.8u_1(t-1) - 0.5u_2(t-1) \\ y_2(t) = & \dfrac{y_1(t-1)y_2(t-1)}{1+y_2^2(t-1)} + 0.2y_2(t-1) \\ & +0.4u_1(t-1) + 0.4u_2(t-1) \end{array}\right\} \qquad (26)$$

The newly proposed control scheme was employed for the system (26), and the control result after 5,000 training iterations is summarized in Figure 3, and results of calculated PID gains are shown in Figure 4. Here, the fixed PID gains $K_{P\,i}^F, K_{I\,i}^F, K_{D\,i}^F$ and precompensator H are designed the same as follows:

$$\left.\begin{array}{l} K_{P1} = 0.147,\ K_{I1} = 0.302,\ K_{D1} = 0.0 \\ K_{P2} = 0.663,\ K_{I2} = 0.548,\ K_{D2} = 0.0 \end{array}\right\} \qquad (27)$$

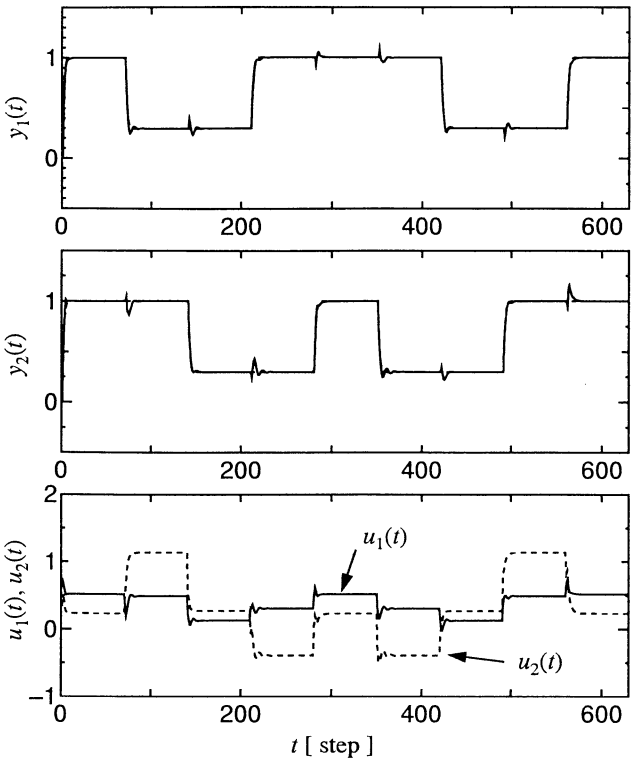


Figure 3: Control results by using the newly proposed control scheme.

$$H = \begin{bmatrix} 0.425 & 0.546 \\ -0.690 & 1.260 \end{bmatrix}. \tag{28}$$

Furthermore, the numbers of the units in the input and the hidden layer, q and r, are set to 11 and 5, respectively. According to the proposed control scheme, it is found that interactions can be removed and good tracking property for the reference signals can be obtained. Furthermore, in Figure 4, it can be seen that PID gains change suitably for each reference signal.

From these results, it is clear that the newly proposed control scheme works well.

4 Conclusions

In this paper, a design method of multivariable neuro-controllers with PID structure was proposed. Furthermore, the effectiveness of the proposed scheme was illustrated through a numerical simulation example. We are now in employing it to a multivariable process control system.

References

[1] Zieglar JG, Nichols NB (1942), Optimum Setting for Automatic Controllers. Trans.ASME, 64(8):759-768

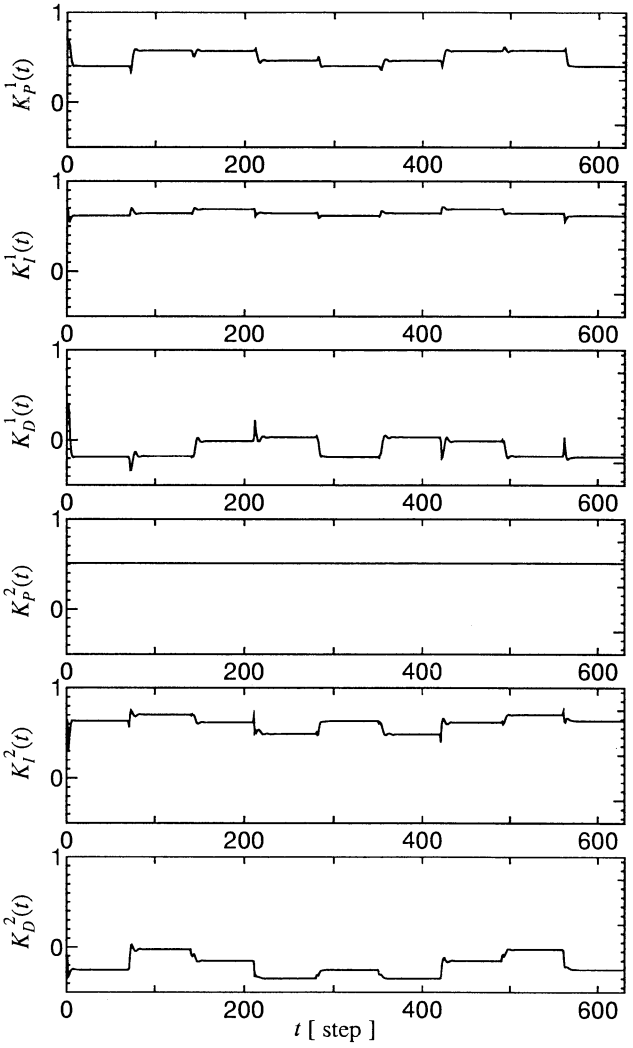


Figure 4: PID gain trajectories used in Figure 5.

[2] Chien KL, Hrones JA, Reswick JB (1972), On the Automatic Control of Generalized Passive Systems. Trans.ASME, 74:175-185

[3] Yamamoto T, Shah SL (1998), Design and Experimental Evaluation of a Multivariable Self-Tuning Controller. Proc.of IEEE Conference on Control Application, Trieste, pp.1230-1234

[4] Shah SL (1972), Design Studies of Multivariable Control Systems. M.Sc.Thesis, UMIST, U.K.

[5] Haykin S (1994), Neural Networks. Macmillan College Publishing Company

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Position Control of Hard Disk Driver by Intelligent Approach

Toru Fujinaka, Michifumi Yoshioka, and Sigeru Omatu
Dept. of Computer and Systems Sciences
Osaka Prefecture University
Sakai, Osaka, 599-8531 Japan
fuji@cs.osakafu-u.ac.jp

Abstract

A new self-tuning neuro-PID controller is proposed and applied to the position control of magnetic head in a hard disk drive. The PID gains are tuned using two types of neural network, which reduces the burden of trial and error by human operators. Simulation results show the effectiveness of the proposed method over various conventional control methods.

Key words: intelligent control, self-tuning, neuro-PID, hard disk drive.

1 Introduction

PID controller has long been used as a major control method, due to its simple structure and robustness against noise and parameter variations. However, the operators must select suitable PID gains from time to time, based on their experience and knowledge on the plant dynamics, in order to attain the robustness.

Recently, intelligent control approaches like fuzzy theory, neural network, genetic algorithm, artificial life, etc. have been developed and they have been applied to many real control problems. In this paper we propose an auto-tuning method of the PID gains using neural networks, and apply this to the position control of hard disk drive.

The magnetic head of a hard disk drive must read the data stored on the track of the disk through magnetic head which is fixed on the arm. Changing the current of the voice coil motor controls the position of the arm [1]–[3]. The driver position follows the center of the track even if the track will move according to the deviation of the disk. Generally, the track of the disk will rotate at 3,600 rpm or 7,200 rpm and the track density is high. Hence, performance of the hard disk drive becomes sensitive to deviation of the disk. We will show that a good tracking performance is attained with the proposed self-tuning PID controller, without a priori knowledge of the precise parameters of the controlled plant. We also compare the results with those obtained using other control methods.

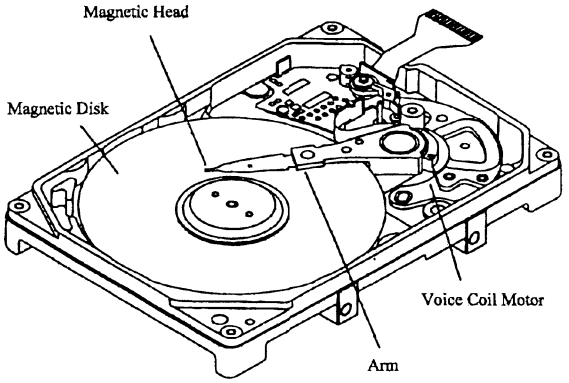


Fig. 1 Hard disk drive

2 Hard Disk Drive System

We consider the hard disk drive shown in Fig. 1, where the voice coil motor is the actuator of the arm with a magnetic head at the end.

The dynamics of the hard disk drive is approximated by a second order linear discrete-time system

$$\begin{aligned} x(n+1) &= Ax(n) + Bu(n) \\ y(n) &= Cx(n) \end{aligned} \tag{1}$$

where $u(n)$ is the input current to the voice coil motor and $y(n)$ is the position of the magnetic head.

The coefficients of the above equation have been obtained by sampling the continuous-time model with the sampling frequency of 6kHz. This has been derived from the fact that the hard disk rotates at 3,600 rpm

(=60Hz) and each track has 100 sectors. The aim of the control is to move the head onto the specified track. Each track is ideally located on a circle, but it actually becomes elliptic, due to the high storage density and downsizing of the hard disk causing off-centered rotation.

Thus, the head position must follow the cyclic motion at 3,600 rpm. Furthermore, the disk has specific resonance frequency about 2–3 kHz due to the mechanical vibration. Therefore, the control objective has the following three specifications:

1) Following to the cyclic motion (servo problem)

2) Stable tracking (robust problem)

3) Reduction of the internal vibration

Conventional control scheme, such as simple feedback, adaptive control, linear quadratic regulator, applied to this problem is not satisfactory, since each of them does not satisfy one or more of the specifications. A controller based on H^∞ control theory has a potential of satisfying all the specifications, but it is not easy to obtain a good description of the nominal plant together with the range of uncertainties.

We apply a control algorithm based on the Proportional+Integral+Derivative (PID) controller that has been adopted to various application problems. The PID controller has simple structure, and the operators can change its parameters according to various environments skillfully, since they understand the dynamic effect of PID parameter changes.

3 Neuro-PID Control Systems

A control system using the neuro-PID controller, a PID controller whose gain parameters are tuned using a neural network, is illustrated in Fig. 2.

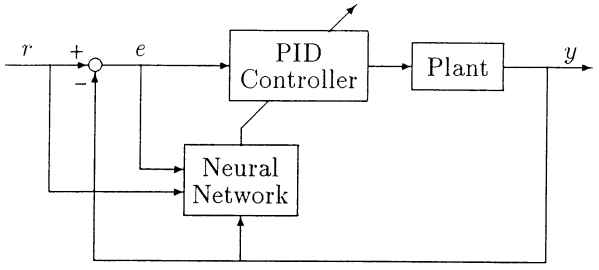


Fig. 2 Neuro-PID control system

In Fig. 2 the neural network adjusts the PID gains, just like a human operator, based on the input u, output y, and the desired signal r, such that the resulting output error may decrease as much as possible. As mentioned before, we must consider the three specifications to design the PID controller. The first one is described by

$$\min_{K_P, K_I, K_D} J_1, \quad J_1 = \frac{1}{2}e(n+1)^2.$$

The second one is to design the controller such that the stable motion can be realized, which can be given by

$$\min_{K_P, K_I, K_D} J_2, \quad J_2 = \frac{1}{2}\frac{x_2(n+1)^2}{1+e(n+1)^2}.$$

Here, $x := [x_1 \ x_2]^T$, and x_2 represents the head velocity. The third one is to reduce the internal mechanical vibration around 2–3 kHz, which can be described as

$$\min_{K_P, K_I, K_D} J_3, \quad J_3 = \frac{|G_{open}(z_0)|}{\gamma},$$

where γ denotes a positive normalizing constant and $G_{open}(z_0)$ denotes the open loop transfer function at $z = z_0$. The open loop transfer function $G_{open}(z_0)$ is given by

$$G_{open} = \left(1 + \frac{K_I}{z-1}\right) G_p(z) G_c(z), \tag{2}$$

where $G_p(z)$ denotes the plant transfer function and $G_c(z)$ is the controller transfer function given by

$$G_c(z) = \frac{K_P(1-z^{-1}) + K_I + K_D(1-2z^{-1}+z^{-2})}{1-z^{-1}} \tag{3}$$

The neural network used here is three-layered type, where the input consists of the past plant output, past plant input, and the desired signal. The output of the neural network consists of PID gains. The structure of the layered neural network is illustrated in Fig. 3.

Using the error back-propagation method, we can derive the recursive algorithm to adjust the PID gains. The detail for derivation of this algorithm has been reported in [4], except that the cost function J is modified as

$$J = J_1 + \gamma_1 J_2 + \gamma_2 J_3. \tag{4}$$

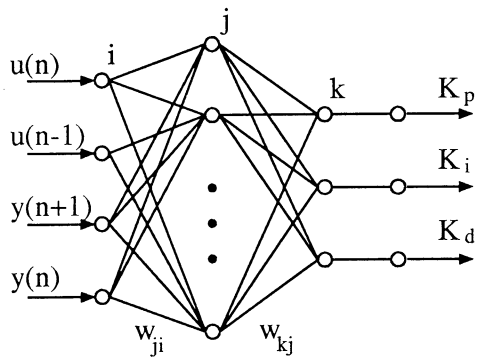


Fig. 3 Neuro-tuning scheme

The basic forms of the adjustment rules are

$$\begin{aligned} \Delta w_{kj}(n+1) &= -\eta \frac{\partial J}{\partial W_{kj}} + \alpha \Delta w_{kj}(n) \\ \Delta w_{ji}(n+1) &= -\eta \frac{\partial J}{\partial W_{ji}} + \alpha \Delta w_{ji}(n) \end{aligned} \tag{5}$$

with $\eta > 0$. Using the above relation, we can adjust the PID gains and the desired output of the head position can be achieved.

4 Simulation Results

We assume that the unknown plant parameters take the following value:

$$A = \begin{bmatrix} 0.9987 & 0.0002 \\ -15.6908 & 0.9088 \end{bmatrix}, \quad B = \begin{bmatrix} 0.7096 \\ 8381.7 \end{bmatrix} \tag{6}$$
$$C = [1 \quad 0]$$

We will consider the situation with the following parameters:

$$\begin{aligned} &z_0 = 2\pi \times 2000, \quad \eta = 6 \times 10^{-7} \\ &\alpha = 1 \times 10^{-7}, \quad \gamma = 5 \\ &r(n) = \sin(2\pi \times 60n) \end{aligned}$$

The remaining parameters are determined based on the simulation results. Figure 4 shows the time response for several values of γ_1, while Fig. 5 shows the frequency for various values of γ_2.

From these results we have chosen the parameters as

$$\gamma_1 = 5 \times 10^{-4}, \quad \gamma_2 = 1 \times 10^4.$$

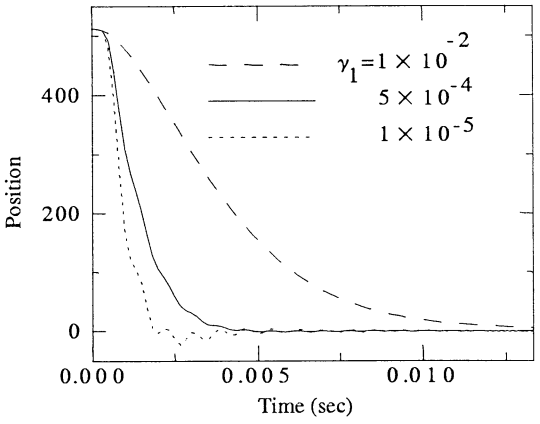


Fig. 4 Time response

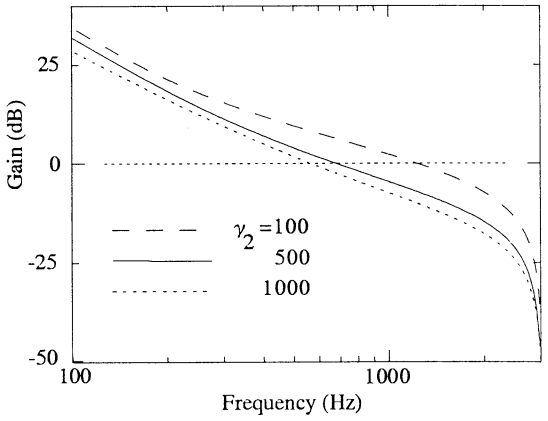


Fig. 5 Frequency response

With these parameters we performed a simulation with sinusoidal reference input. The transient trajectories are illustrated in Fig. 6, where the response is compared to those obtained by a few other methods.

The generalized plant we used for designing the H^∞ controller is shown in Fig. 7, where the weighting functions $W_s(s)$ and $W_a(s)$ are chosen to be

$$W_s(s) = \frac{2500}{s + 200\pi}, \qquad W_a(s) = \frac{5s}{s + 2000\pi}. \tag{7}$$

Figure 8 shows the steady state errors for various methods, and Fig. 9 shows the corresponding gains characteristics.

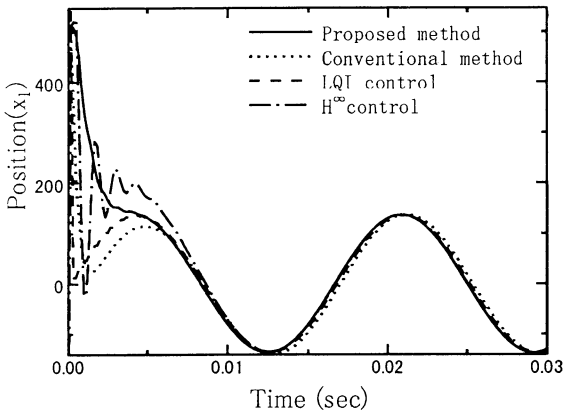


Fig. 6 Transient responses for various methods

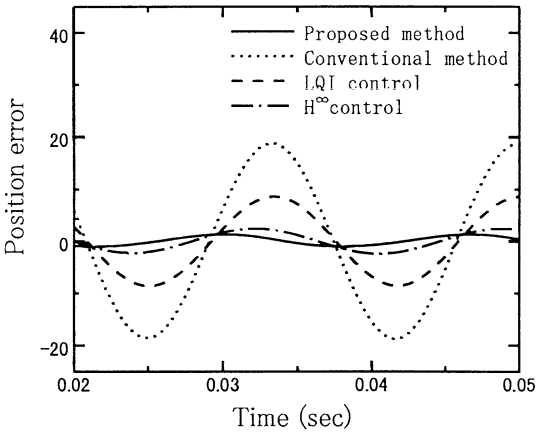


Fig. 8 Tracking errors for various methods

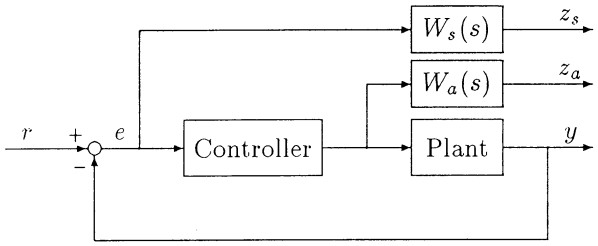


Fig. 7 Generalized plant for H^∞ control

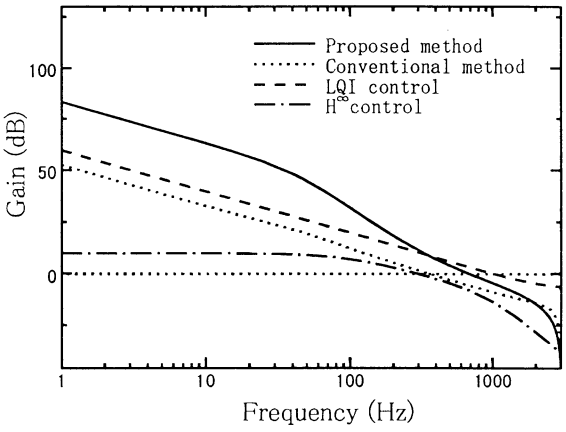


Fig. 9 Frequency response for various methods

Here, the conventional method means the one that has been adopted for the hard disk controller and LQI is the integrated linear quadratic control algorithm, which is effective when the desired signal is constant. In Fig. 8, the steady state error of the proposed method is smaller than that by H^∞ controller and from Fig. 9 the proposed method is more effective than H^∞ controller, since the present method has produced a higher gain in the low frequency range and lower gains in the effective frequency band for internal mechanical vibration. From these results we can see that the proposed method has the better control results compared with the other methods.

5 Conclusions

In this paper we proposed a new technique for tuning the PID controller gains which can be applied to the position control of the magnetic head of a hard disk drive. The structure of the resulting controller is simple, yet shows a better control performance over those controllers using conventional techniques.

References

[1] T. Mita, *H^∞ Control*, Syokodo, Tokyo, 1994 (in Japanese).

[2] M. Tomizuka, T. Chin, and C. Kia, "Analysis and Synthesis of Discrete-Time Repetitive Controllers", *ASME J. Dyn. Syst. Meas.Control*, Vol. 111, pp. 353–358, 1989.

[3] M. Kobayashi, T. Yamaguchi, et al., "Adaptive Control of a Hybrid Servo System for Magnetic Disk Drives", *JSME Int. J.*, Series C, Vol. 39, No. 4, pp. 772–780, 1996.

[4] S. Omatu, K. Marzuki, and Y. Rubiya, *Neuro-Control and Applications*, Springer, London, 1995.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Application of Fuzzy Behavior-Based Control for a PUMA Robot

Palitha Dassanayake*, Keigo Watanabe**, Kazuo Kiguchi**, and Kiyotaka Izumi***
*Graduate School of Science and Engineering,
Faculty of Engineering Systems and Technology,
**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
†E-mail: watanabe@me.saga-u.ac.jp

Abstract

A fuzzy realization of a behavior-based control system is applied to six-degree-of-freedom, six-link manipulator. The control system that had been applied to a mobile robot in a previous work, is modified to suit to the PUMA robot. Fuzzy behavior elements are trained by a genetic algorithm. Apart from the basic concept, a servo controller is also introduced between the fuzzy controller and the robot so that the outputs from the fuzzy controller are distances and orientations in the Cartesian space. Result shows that fuzzy behavior based control approach can be applied to multi-link manipulators to reach a particular point while avoiding an obstacle.

Keywords: PUMA robot, Multi-link manipulator, Fuzzy behavior-based control, Obstacle avoidance

1 Introduction

R. A. Brooks [1] introduced a new architecture called "Subsumption Architecture" for controlling mobile robot. This behavior-based control has been actively applied to artificial Intelligent robots [2]-[4]. On the other hand, on-line collision avoidance methods have been introduced for manipulators, but they are not based on behavioral approach [5], [6]. Therefore in this work, it has been tested for a six-degree-of freedom, six-link PUMA robot to reach a target while avoiding obstacles based on fuzzy behavioral approach.

2 The Manipulator, Obstacles and the Target

The robot has six-degree-of freedom and it moves in the three dimensional space. The link axes are selected as shown in **Fig. 1**. The dynamic equation for the manipulator is given in the matrix-vector form by

$$\boldsymbol{\tau} = D(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \boldsymbol{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \boldsymbol{g}(\boldsymbol{\theta}) \qquad (1)$$

where $D(\boldsymbol{\theta})$ is the 6×6 inertia matrix, $\boldsymbol{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$ is the 6×1 Coriolis and centrifugal force vector, and $\boldsymbol{g}(\boldsymbol{\theta})$ is the 6×1 gravitational vector. $\boldsymbol{\tau}$ is the torque vector and $\boldsymbol{\theta}$, $\dot{\boldsymbol{\theta}}$ and $\ddot{\boldsymbol{\theta}}$ are the joint angle vector, joint angular velocity vector and the joint angular acceleration vector respectively.

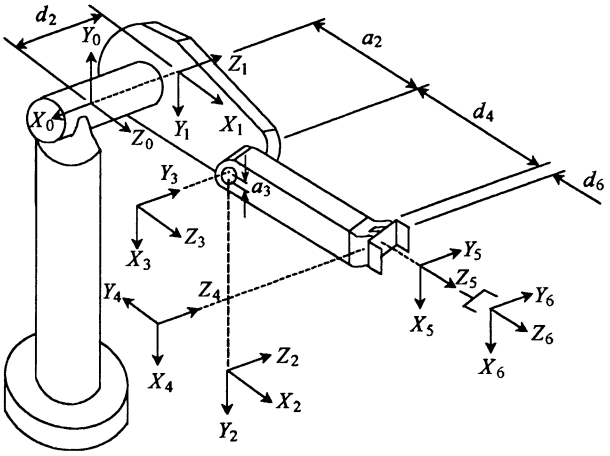


Figure 1: The PUMA robot manipulator and the coordinate axes

The robot is simulated by computing the actual dynamics numerically with Newton-Euler formulation [7] and no assumptions are made to simplify the dynamics of the system, but the friction is neglected.

The target is assumed to be a small sphere with a radius 5[mm] and it is placed on (−0.9, 0.0, 0.0). The obstacle is assumed to be at (−0.5, 0.0, 0.5) with a diameter of 180[mm]. Since the thickness of the links is not considered, 10 [mm] allowance or safety factor is allocated around the obstacle. Therefore, in the simulation it is assumed that the diameter of the obstacle is 200 [mm]. Initially, the robot's joint angles vector is (90, −90, 0, 0, 0, 0) measured in [deg] and the end-effector point coordinate is (−0.14909, −0.02032, 0.92112). The above mentioned coordinates are measured in [m] with repsect to the base coordiante system. The obstacle is selected in such a way

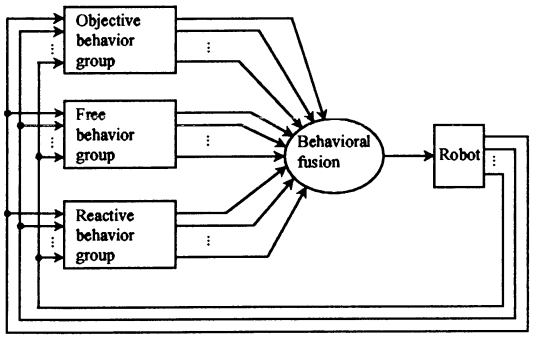


Figure 2: Behavior based control system

that if a straight line is drawn from the initial end-effector point to the target, the line goes through the obstacle. In the simulation the objective is to minimize the total travel distance of the end-effector point to an acceptable solution when reaching the target while avoiding the obstacle.

3 Concept

Figure 2 shows an example of the behavior-based control system consisting of three behavior groups for a robot with the higher level behavior group shown over the lower behavior group. The lowest level group or the reactive behavior group basically consists of some behavioral elements that detect or avoid obstacles. The free or adaptive behavior group is composed of some behavioral elements that move or detect a free space or move freely. These behavioral elements connect with a higher level behavior. The objective behavior group consists of some behavioral elements that search or track a target, which is an essential purpose (or task) of the robot. The behavioral fusion part fuses the behavior results generated from each behavior group through competition or cooperation. It should be noted that this figure is persistently a basic concept for the behavioral element. In this work, each behavior group consists of six inputs and each input is connected to an output by a fuzzy relationship and there are no other interconections between the inputs and the outputs. The input vectors of objective behavior group are the distance vector from the end-effector point to the target, $\boldsymbol{D} = [D_x\ D_y\ D_z]^T$, and the relative angle vector between the moving direction of the end-effector point and the objective point, $\Psi = [\Psi_x\ \Psi_y\ \Psi_z]^T$, with respect to the base coordinate system. The free behavior group input vectors are the velocity vector, $\boldsymbol{v} = [v_x\ v_y\ v_z]^T$, and the angular velocity vector, $\boldsymbol{\omega} = [\omega_x\ \omega_y\ \omega_z]^T$, of the end-effector point with respect to the base coordinate system.

It is assumed that the center coordinate and the radius of the obstacles can be obtained by a processed CCD camera in absolute coordinates. If d, is the minimum distance of the nearest obstacle from all the link, the inputs to the reactive behavior group are, the distance vector along d from the link to obstacle, $[d_x\ d_y\ d_z]^T$, with respect to the base coordinate system, and the relative angle vector of the obstacle to the minimum point on that link, $[\psi_x\ \psi_y\ \psi_z]^T$ with respect to the base coordinate system. They can be calculated if the current joint angular positions are known with the obstacle center coordinate and the radius.

The outputs are the desired position vector and the desired orientation vector of the end-effector point for each behavior group. The orientation is given by the angle measured from the axis of rotation of the end-effector point with respect to the based coordinate system.

Each behavior group outputs are as follows. $[x_{eod}\ y_{eod}\ z_{eod}]^T$, $[x_{efd}\ y_{efd}\ z_{efd}]^T$ and $[x_{erd}\ y_{erd}\ z_{erd}]^T$ are the desired position vectors, and $[\alpha_{ox}\ \alpha_{oy}\ \alpha_{oz}]^T$, $[\alpha_{fx}\ \alpha_{fy}\ \alpha_{fz}]^T$ and $[\alpha_{rx}\ \alpha_{ry}\ \alpha_{rz}]^T$ are the desired orientation vectors of the end-effector point for the objective behavior group, free behavior group and the reactive behavior group respectively.

The desired position $[x_{ed}\ y_{ed}\ z_{ed}]^T$ and desired orientation $[\alpha_{ed}\ \alpha_{ed}\ \alpha_{ed}]^T$ of the end-effector point with respect to the base coordinate system are obtained by fusing the reasoning results generated from each behavior group. Fuzzy rule relation and fusion of behavior group are described in next sections.

Apart from the basic concept, a servo controller is introduced between the fuzzy controller and the robot so that fuzzy controller outputs are the desired positions and orientations in the Cartesian space. At each sample interval the outputs from the fuzzy controller are fed to the servo controller. The servo controller calculates the desired torque outputs for the given Cartesian position and orientation, and the outputs from the servo controller are applied to the robot.

3.1 Fuzzy rule relation

The resultant fuzzy reasoning consequent is obtained as

$$u^* = \sum_{i=1}^{M} p_i w_{bi} \tag{2}$$

where M denotes the total number of rules. w_{bi} the constant in the conclusion of the ith rule, and p_i the normalized rule confidence such as

$$p_i = \frac{\mu_i}{\sum_{j=1}^{M} \mu_i} \tag{3}$$

$$\mu_i = \exp\{(\ln(0.5)(y - w_{ci})^2 w_{di}^2\} \tag{4}$$

Here, w_{ci} denotes the center value associated with the ith membership function, and w_{di} denotes the reciprocal value of the deviation from the center w_{ci} to which the ith Gaussian function of the input data on the support set has value 0.5.

3.2 Fusion of behavior groups

Let the two same outputs from two behavior groups be described as a and b from behavior group i and behavior group $i+1$ respectively. The fusion result is given by c where, c is calculated as follows.

$$c = (1 - s)a + sb \tag{5}$$

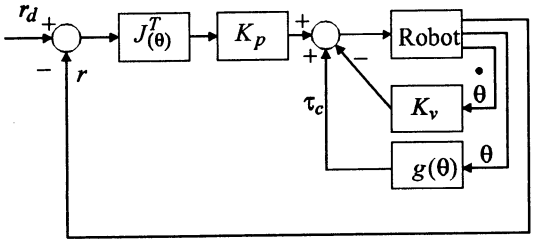


Figure 3: The servo controller

$$s = |\mathrm{sat}(a) + \mathrm{sat}(b)|/2 \tag{6}$$

$$\mathrm{sat}(x) = \begin{cases} \mathrm{sgn}(x) & |x| > \varepsilon \\ x/\varepsilon & |x| \leq \varepsilon \end{cases} \tag{7}$$

where ε denotes a positive number.

In this work, the reactive behavior group is the lowest behavior group and the objective behavior group is the highest behavior group. The behavioral fusion is first taken place with the objective and free behavior groups and thereafter the fusion results is again fused with the reactive behavior group in order to get the final outputs of the fuzzy controller.

4 The Servo Controller

In reference [8], it is concluded that in control stability of the PUMA-560 robot manipulator the proportional-derivate (PD) controller is more robust than the proportional-integral-derivate (PID) and in reference [9] they have found that Cartesian servo algorithms which use the transpose of the Jacobian and model-based gravity compensation, provide easy programmability and accurate reproduction of stiffness over a wide range. Therefore a PD servo controller is used, which is shown in **Fig 3**. In the figure, $\boldsymbol{r}_d$ and $\boldsymbol{r}$ are the desired reference Cartesian position and orientation vector and the actual Cartesian position and orientation vector, and the Jacobain transpose matrix J^T is used to transform the Cartesian position and orientation difference $(\boldsymbol{r}_d - \boldsymbol{r})$ into joint angular position difference. K_p and K_v are the position gain and the velocity gain matrices and $\boldsymbol{\theta}$ is the joint angle vector and $\dot{\boldsymbol{\theta}}$ is the angular velocity vector. $g(\boldsymbol{\theta})$ vector is the torque required to compensate gravity.

K_p and K_v matrices are selected in such way that they satisfy the stability conditions and the torque outputs from the servo controller are in the normal working range of the robot. In order to satisfy the stability condition, K_p and K_v matrices must be positive definite.

The K_p is found to be

$$\begin{bmatrix} 15 & 0.05 & -0.57 \\ 0.05 & 9.7 & 1.8 \\ -0.57 & 1.8 & 1.6 \\ 0.08 & -0.00021 & -0.00018 \\ -0.00022 & 0.0098 & 0.008 \\ 0.00015 & 0.0000053 & 0.0000053 \end{bmatrix} \begin{bmatrix} 0.008 & -0.00022 & 0.00015 \\ -0.00021 & 0.0098 & 0.0000053 \\ -0.00018 & 0.008 & 0.0000053 \\ 0.072 & 0.0000065 & 0.00017 \\ 0.0000065 & 0.0028 & 0.0 \\ 0.00017 & 0.0 & 0.00017 \end{bmatrix} \tag{8}$$

and $K_p = 3K_v$.

Eigen values of the K_p and K_v are also calculated and all of them are positive and hence matrices are positive definite. Therefore the system is stable.

5 GA Parameters

One individual has 270 parameters. Number of individuals in one generation is 60. Number of parents selected is 60. Parents are selected from the tournament selection and the same individual can be selected as a parent more than once. Crossover probability is 0.6. Mutation rate is 1/(code length). All offspring and parents are placed into a pool and a score is given to each individual from the randomly selected other 10 individuals. If the fitness value of the individual is higher than the randomly selected individual then score is increased by one, otherwise it is the same. Initial score of each individual is zero. From that score the better 60 individuals are selected for the next generation.

5.1 Fitness function

The manipulator's one individual run is over if, one or more links go out of range, or any of the link collides with the obstacle, or the given time is over, or the robot end-effector point moves 0.1[m] away from the minimum distance of the end-effector point and the target, or the end-effector point reaches the target successfully. Fitness value Fit is given by

$$Fit = 3.0 - d_{\min} + (D_0 - d_{\min})/T_D - 2.0 \times N_o - 0.2 \times N_c \tag{9}$$

where $d_{\min}$ is the minimum distance of the robot's end-effector point and the target during the run, D_0 is the initial distance between the end-effector point and the target and T_D is the travel distance of the end-effector point during the period of that run. Here N_o and N_c are integers where N_o is equal to the number of links went out of range and N_c is equal to the number of links collided with the obstacle.

6 Simulation and the Result

The end-effector point is to start from the starting point and go to the target. The fittest individual's manipulator path after 30 generations is shown in **Fig. 4**

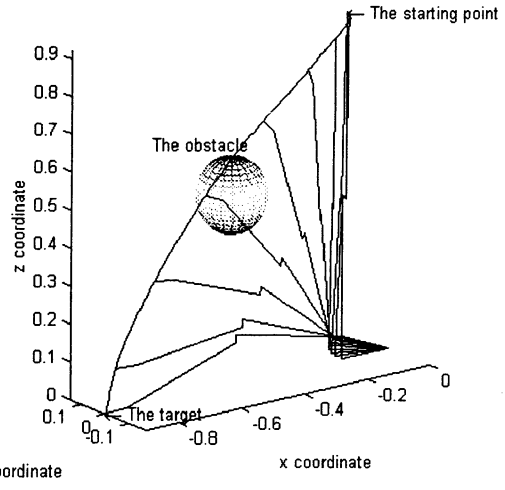


Figure 4: The fittest individual's path after 30 generations in one view

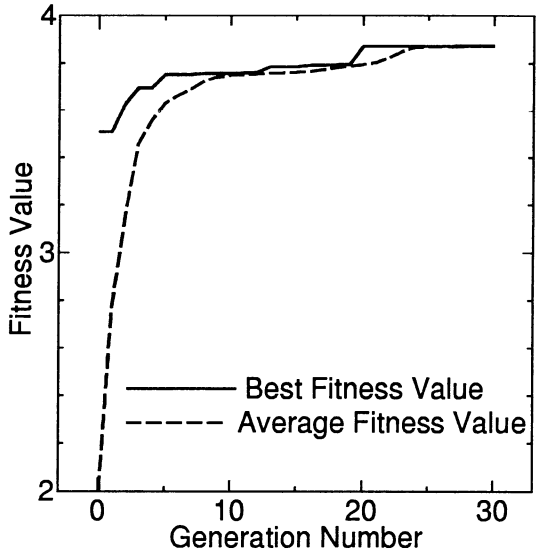


Figure 5: The best and average fitness values

and it shows that the robot managed to reach the target while avoiding the obstacle in a reasonable manner. The fitness value of this individual was 3.850005 The best fitness value and the average fitness value from 0 to 30 generations are shown in the **Fig. 5**.

7 Conclusions

It has been proved from the simulation that the proposed method is effective for complex manipulators like PUMA robot to achieve certain tasks. Use of fuzzy behavior-based strategy had the advantage of moving towards a particular point without knowing the inverse kinematics. It implies that the manipulator can be controlled to achieve the desired task with a suitable fitness function without going into actual analytical details. This is very useful especially in manipulators with higher degree-of-freedom because the kinematics and dynamics of such manipulators are complex by nature.

Irrespective of the strategy used, though we have used the fuzzy behavior-based strategy in this work, it can be concluded that in motion control, it is easier to control the manipulator by separating the control system into two parts such that one is to find the desired motion and another part to calculate the required torque for the motion.

References

[1] R.A. Brooks, "A Robust Layered Control Systems for a Mobile Robot," *IEEE Journal of Robotics and Automation*, vol. 2 no. 1, pp. 14–23, 1986.

[2] P. Maes, "Learning Behavior Networks from Experience," in *Procs. of the First European Conference on Artificial Life*, 1992, pp. 48–57.

[3] J.R. Koza, "Evolution of Subsumption Using Genetic Programming," in *Procs. of the First European Conference on Artificial Life*, 1992, pp. 110–119.

[4] K. Watanabe and K. Izumi, "Construction of Fuzzy Behavior-Based Control Systems for a Mobile robot," in *Procs. of the Third Int. Symposium on Artificial Life and Robotics (AROB 3rd'98)*, 1998, vol. 2, pp. 518–523.

[5] N. Rahmanian-shahri and I. Troch, "Collision-Avoidance for Redundant Robots through Control of the Self-Motion of the Manipulator," *J. of Intelligent and Robotic Systems*, vol. 16, no. 2, pp. 123–149, 1996.

[6] H. Ding and S.P. Chan, "A Real-Time Planning Algorithm for Obstacle Avoidance of Redundant Robots," *J. of Intelligent and Robotic Systems*, vol. 16, no. 3, pp. 229–243, 1996.

[7] K.S. Fu, R.C. Gonazalez, and C.S.G. Lee, *Robotics, Control, Sensing, Vision, and Intelligence*, New York, NY: McGraw-Hill, 1987, pp. 103-118.

[8] K.P. Valvanis, T.B. Larsson, and S.P. Gardner, "PD and PID Model-Based Control Stability Analysis of the PUMA-560 Robot Manipulator under Model Mismatch," *J. of Intelligent and Robotic Systems*, vol. 7, no. 2, pp. 233–254, 1993.

[9] J. Fiala and A.J. Wavering, "Experimental Evaluation of Cartesian Stiffness Control on a Seven Degree-of-Freedom Robot Arm," *J. of Intelligent and Robotic Systems*, vol. 5, no. 1, pp. 5–24, 1992.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

DNA Computing --- A Novel Paradigm of Artificial Life *in vitro*

Katsunori SHIMOHARA and Jian-Qin LIU

ATR Human Information Processing Res. Labs., Evolutionary Systems Department,
Soraku-gun Seika-cho, Hikaridai 2-2, Kyoto, 619-0288, JAPAN

E-mail: {katsu,jqliu}@hip.atr.co.jp

Abstract

DNA computing has become one of the most promising ways to build the next generation of computer systems since Leonard Adleman initiate the new discipline in 1994. Up to now DNA computing has includes three major branches ---- formalization for DNA computation , wetware construction with biological media, and computational simulation by conventional tools. As a kind of formal description, abstract machines play an important role with respect to the computational power of DNA computing. The grammar representation of DNA computation ranges within the Chomsky spectrum and corresponding automata is carried out under the constraint of the Turing machine. With DNA molecules, evolution in vitro and computation in the test tube environment have shown a stronger ability in parallel processing and emergent computing owing to the underlying complexity mechanism. Different prerequisites are necessary in molecular implementation of some prototypes of DNA computers and professional skills often influence the efficiency of the corresponding experiments. Furthermore, applications such as gene finding and function prediction also provide much more space for the intersecting field of DNA computation --- an example of artificial life *in vitro* --- and efforts to explore the principle of living system in the nano-level. DNA computers can surpass some limits of traditional hardware and it can also be simulated in terms of demonstration *in info*. As we predicate in this survey, some progresses in the aspect of improving throughputs of DNA computers will be made, based on embedding genome informatics in wetware architecture and controlling the molecular fabrication processes.

Keywords: DNA computer, bioinformatics, functional genomics, artificial life, nano-computing

1.Introduction

DNA computation initiated by Prof. L. Adleman, is a kind of unconventional way to build parallel distributed systems with the power to massively process information, in terms of molecular forms of DNA and corresponding operations on the biological media among a test tube, rather than semiconductor. It also provides an alternative for the architecture of nano-computers or molecular quantum computers. DNA computing is a new method for building computer systems. But briefly speaking, its advantages over other techniques lies in the underlying mechanism for parallel computing and its ability in solving NP problems. Further more, the problem, which can be described as:

considering the merits mentioned above, how can we say that this model can replace the conventional computer?

Still remains open. Of course, there exist these kinds of probabilities. But the majority of existing techniques have been proven to be equivalent to the Turing machine. Maybe somebody will say:

ah, the existing techniques such as electrical computer systems since the ENIAC, do the job like this way. You can not do more than that, so we can not expect your scheme to be a real alternative one for the traditional solution

So, we have to find a stronger one in computational power --- fortunately we have found ---- it is the Genome Dynamics (we call it so). Now we are proving it to be a Super-Turing machine based on our DAN architecture.

Briefly speaking, a Super-Turing machine means a computational machine that has a stronger ability than Turing one.

Besides the definition mentioned above, in the view of the theoretical computer science, it means that a string is used as input to the Turing machine and Super-Turing machine. But the results are totally different, i.e., the former can not handle it and can not stop this process owing to the underlying mechanism within it. Furthermore the latter will stop after the string has been parsed or generated according to the criterion concerned, it shows that the internal mechanism in which some processes of parsing or generating are available, exist in essential.

Besides in the view of the theoretical computer science --- what we talk about is put in the framework of the abstract machine, several levels exist when we begin to make a computer. Of course the Super-Turing machine we desired to make, belongs to one of the computer with a generalized meaning. So, how to make it from the most abstract level to a concrete form, will become one of the crucial steps in our work.

Induced by the abstract form, a DNA computer can be built as the following ways:

(1) Simulation model and software, e.g., our Genetyllis system;
(2) A prototype that is based on (1) and can be used to demonstrate the principle of the new style of the computing paradigm;
(3) The architecture for DNA computer from design to implementation of wetware;
(4) Effort to solve some typical real world problems that are more practical than the theoretical examples such as Hamilton path problem.
(5) Applications start to be found by the DNA researchers.

2. Key factors in DNA computing

2.1 Theoretical considerations

The DNA schemes are expected to solve the following problems in our work:
(1) Theoretical problem:
Logic is important since it has been regarded as a basic function in computer systems, so is it for DNA implementation of logic units.
(2) Computer number system.
A new number system is necessary to be built for the whole system.

2.2 Practical Strategies

When we finish the theoretical preparation mentioned above, we can begin to consider the implementation that are related to the following factors:
(1) Our GD model (GD = genome dynamics, here the notation is used only for convenience);
(2) The relation between our GD model and the DNA model;
(3) The architecture of DNA computers;
(4) The potential form for the experiment concerned.
(5) Applications (with DNA chips and DNA micro-array for functional prediction).
The features such as emergence, meta-evolution, emergence of meta-evolution, evolutionary dynamics, genome dynamics that is defined as evolutionary dynamics of genome, are necessary in the concerned modeling procedure. Obviously, GE itself is not a DNA computing paradigm, but its relation with DNA computing should be owned to the mechanism for information representation with the genomic symbols. The following points can support the links between them when a DNA computer is built:
At first, GD has the merits in the theoretical aspects above (e.g. computational power), we can expect it to provide a kind of new computational capacity. Its basis is stronger in the meaning of computation.
Secondly, GD is a process that can handle, generate and control the symbol string for (artificial) genome.
Thirdly, GD can be used to derive the prototype of DNA computers, where the sharp differences between the GD model and other ALife models can be observed. The symbolic representation plays an important role here and its structure reflects some enlightenment from the real genome so that GD is not a random mechanism. It is nonlinear and deterministic form. But it can not be predicted for long term, rather than statistical mechanisms for programming processes.
Because we do not want the DNA computing to become a time-consuming process, it is imperative to avoid the linear-ruled way and try to find some tools that are feasible in NP cases.

2.3 Empirical factors

To avoid the factor of randomness both in computing and in bio-chemical reaction, we select a nonlinear mechanism rather than randomness for our Genetyllis system.
More empirical arrangements are needed when a prototype of DNA computers is made. Starting from the simulation for "wet-CPU" in our genetyllis system, the whole structure is arranged as a circular hierarchical structure and underlying mechanism. We adopt the solution that we should define the operator and operation as a different way compared to the hardware CPU. The structure designed here is enlightened by functional structure in genomic representation and also is a kind of optimization one in the information geometry (topology).
Then we can make the whole system, in which the "memory-like" module and compiled code will be embedded into the same structure. In a word, the DNA computer we want to build is the one in which the maximum amount of the information with respect to the capacity in evolutionary processes is reflected by imitating the features in real genomes.
About the granular in designing the simulation prototype for the wetware architecture, the procedure can be defined as the following steps:
Firstly, a media that can store and code the genome that are regarded as the "wet-program" here, and the domain constraints such as ORF, alignment, phylogenetic tree and others are used as the control mechanism for the (en) coding and decoding. However, more important is some hint from protein interaction modeling for guidance of the "running" procedure. Secondly, the major units are arithmetic and logic units that need to be described as the kinds of specification diagrams. Thirdly, checking what kind of tasks can be undertaken by the paradigm of DNA computers is inevitable. Uncertainty inference for data of pattern and symbolic information (under the condition that incomplete, noisy, faults and etc.) exist in the spatial-temporal domain. This is not limited to the case of the NP because NP is only a problem of time. The factors related with the complexity of the object problem are also not limited to the simply repeating (building) the traditional form and modules of electrical computers such as register, memory, CPUs, counter and etc. by DNA molecules and test

tube tool if they are available there. Finally, real genome sequences can be used as the basis to design the wetware concerned. In essential, this step is carried out on the parallel modeling. The work directed to DNA computers is also consistent with basic intentions in ALife. Briefly speaking, two reasons can be found in those two facts: (a) to explore the parallel processing capacity owing to the huge amount of the genome (gene) sequences, although error reduction scheme should be adopted and bio-chemical techniques needed. But it deserves to be done when we think of the primitive form of the machine that can compute made by another media rather than the traditional chips, and (b) to explore the complexity of the machine should and have to match the problem to be solved.

Since to program the computers is natural, the concept "wet-program" is used to do the tasks as a genomic way (e.g. ORFs are included). Related proteom interpretation for control and guidance is also useful here because the choice for a more powerful computational system with a dynamical way is involved with a kind of dynamical system with a nonlinear feedback mechanism.

Considering the fact that evolution processes are also included in DNA computing as other paradigms in artificial life systems, the factors about "wet-programming" include the mathematical features for the evolution, quantitative measurement and criterion, optimization (global optimization vs. local optimization), population size (e.g. the sufficient large samples), fitness landscape and distribution (e.g. the description made on the non-smooth space) and the data that is different from the typical data structure in conventional software. Of course, the relation between GD, wetware and bio-informatics acts as the kernel among a dynamical form of DNA computers, thoroughly.

3. Wetware Evolution

The evolutions of hardware and software are regarded as two efficient ways to implement an artificial life system. The former is a solution of high speed and real time, the latter is easy to be applied to simulate the instances. Parallel to the two forms --- evolutionary hardware and software, the evolutionary wetware shows its instinct links to the artificial lives in vitro. The principle of evolutionary wetware is very different from the other twos. The models of DNA computing often are defined with the necessary operators. The commonly used set of DNA computing operators can be defined as

Operation Set = {anneal, melt, ligate, polymerase extension, cut, destroy, merge, separate by length, append, mark, unmark}

where the elements within the operation set are assigned with molecular biological meanings.

Among the existing models of DNA computing, the splicing system (H-system) is one of the most popular tools. But the selection and design of formal language description for DNA computing according to the Chomsky's hierarchical classification are always made on the basis of concrete objects in applications.

A wetware consists of five major components --- (a) the formal description, (b) the biological performance, (c) the operation mode, (d) the abstract model and (d) the computational performance. Focussing on the cybernetics units in DNA computers, an example of this kind of prototype is illustrated as the following formulas:

The model = the basic part of standard DNA model
+ context-sensible grammar
+ CA with random-chaotic-syntactic parsing/generating
+ mRNA-enzyme reaction for hetero-species
+ protein verification for DNA computing

The new operation = mRNA-enzyme reaction for hetero-species
+ protein verification for DNA computing

The system = test tube (wetware/simulation)
+ massively parallel computer (functional support from hardware)
+ real experiment for functional genomics (e.g. genefinding for Alzheimer's disease)

The features of evolutionary wetware can be summarized as the four points:

(a) the wetware evolution for biological cellular automata.
(b) the implementation in vitro.
(c) the available support from MIMD machine with massively parallel real-time computing compared to DNA chips. Note that DNA chips ≠ the chips used for DNA computer.
(d) the strong potential in the possible applications such as the medical treatment for human fatal disease, e.g. neurodegenerative disorders such as AD (Alzheimer's), PD (Parkinson's), HD (Huntington's) and WD (Wilson's) [1]).

The feasible reasons can be traced to the situation in information analysis for gene coding, the existing state of operations for DNA molecules, synthesis of complexity systems, pattern recognition with grammar inference and automata generating, formal languages and automata through theoretical computer science, and cellular automata.

When the technical specification is carried out for DNA computational machines, the crucial part of the machine

is the genome and corresponding protoem that are not "digital" but evolved directly in wetware, which satisfies the definition of artificial life with both generalized and core meanings. Compared to the models concerned, architectures have more space for creativity. DNA operation modules to test tube works with the proper molecular biological platform. Some subsystems can be arranged to represent, generate and parse genomes for wetware or through MIMD. The wetware-related units also include (tentative) cellular automata, RNA-guided protein production, functional evaluation unit. External interface is compatible with the architecture of conventional computers, through the way of hardware, software, firmware, groupware, or C/S database systems, in which biological data retrieval for monitoring the communication and cooperation of bio/electrical parts. Communication mechanisms in signal processing involve the aspects such as image, speech, pattern and other information represented as optical, mobile, wireless, sensory and etc. Additional modules such as genefinding devices are oriented to the applications. Note that the bio-cybernetics module is consistent with the core parts with feedback in the system. An example by Genetyllis is given in Figure 1 to show the "wet-program".

4. Conclusion

With the Genetyllis system, some representative patterns have been demonstrated for DNA computers. Our further work is desired to embed it into a network environment for genome/proteom information supporting.

References

[1] Shoulson I (1998), Experimental therapeutics of neurodegenerative disorders: unmet needs, Science, 282:1072-1074.

The genome sequence is generated as

GCTCCACGATAGGCCACCCCGCTCACAACCTCCTCTCTTACAC
AGTTCTCAAGGAGCCCAGACCTTAAGAAACACGTTGAGCCACT
ACCTCACACACAATTGCCAAGCTAATTCAAGTAGACTTCAAGA
TTAACTTTCTTGCCGTGATCAATTCGGAAGCACCTCTACCGGG
ACCTTTCAAGTGCCTACGAGCTCCAACTCCTTTTTGAGATAAG
ACCACGTCTCTAATCTCGACACGAGGCACCACAAAAGTTCCTC
CTGTTAGAGTCACCCCACGGCCTAACCTGAGATACATCCCGA

The two strand (helix) DNA sequence (gene number = 300 x 2) generated as

GCTCCACGATAGGCCACCCCGCTCACAACCTCCTCTCTTACACAGTTCTCAAGGAGCCCA
ATCTTGTAGCGAATTGTTTTATCTGTGGTTCTTCTCTCCGTGTGACCTCTGGAAGATTTG

GACCTTAAGAAACACGTTGAGCCACTACCTCACACACAATTGCCAAGCTAATTCAAGTAG
AGTTCCGGAGGGTGTACCAGATTGTCGTTCTGTGTGTGGCCATTGGATCGGCCTGGACGA

ACTTCAAGATTAACTTTCTTGCCGTGATCAATTCGGAAGCACCTCTACCGGGACCTTTCA
GTCCTGGAGCCGGTCCCTCCATTACAGCTGGCCTAAGGATGTTCTCGTTAAAGTTCCCTG

AGTGCCTACGAGCTCCAACTCCTTTTTGAGATAAGACCACGTCTCTAATCTCGACACGAG
GACATTCGTAGATCTTGGTCTTCCCCCAGAGCGGAGTTGTACTCTCGGCTCTAGTGTAGA

GCACCACAAAAGTTCCTCCTGTTAGAGTCACCCCACGGCCTAACCTGAGATACATCCCGA
ATGTTGTGGGGACCTTCTTCACCGAGACTGTTTTGTAATTCGGTTCAGAGCGTGCTTTAG

Figure 1 An example given by Genetyllis

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

DNA Computing by Genomic Dynamics I: --- Evolutionary Modeling of Emergence and Context-Sensible Grammar Representation

Jian-Qin LIU and Katsunori SHIMOHARA

ATR Human Information Processing Res. Labs., Evolutionary Systems Department,
Soraku-gun, Seika-cho, Hikaridai 2-2, Kyoto, 619-0288, JAPAN

E-mail: {jqliu, katsu}@hip.atr.co.jp

Abstract

Genomic dynamics, whose concepts and tools differ from existing concepts and tools in the domain of artificial life, is defined as the mechanism and corresponding processes of evolutionary dynamics with its complexity and emergence at the genomic level. In this paper, we propose a new model of DNA computation based on genomic dynamics and provide some examples to illustrate the emergent characteristics of dynamical evolution processes.

An evolutionary mechanism of emergence, which we call the (l + m + n) structure, has a higher-dimensional space of the landscape and a higher degree of complexity. From the viewpoint of digital life, we have found a phenomenon not reported to date --- an amount of stable sub-space exists between areas in which emergence can be observed --- owing to the chaotic mechanism accompanying the combinatory nonlinear mapping from the genotype space to the phenotype space. This can be explained as follows --- we have found that in the two-dimensional space, emergence can be observed within non-smooth areas consisting of different separate convex sets.

Along with its dynamical structure in a DNA computational system, a new representation scheme is proposed for describing and parsing DNA sequences. This scheme falls under the category of context-sensible grammar according to Chomsky's hierarchy and an equivalent automaton can be made with the same computational power. An example is given of grammar parsing for the windowed genome sequences of H. pylori, discussions are made on the bio-informatics strategies for reconstructing sub-tree structures in a non-rooted phylogenetic tree. The work reported here is one part of our effort to build a general DNA computing framework in which genome information is applied in a dynamical way.

Keywords: DNA computer, bioinformatics, functional genomics, artificial life, nano-computing

1 Introduction

DNA computation has been highlighted in the community of ALife (e.g. ALIVE VI and ICEC'98) and the Human Genome Project (HGP) is rapidly approaching its expected goal. DNA computation has been regarded as a potential challenging paradigm because of certain achievements such as a solution to the Hamilton path problem by L. Adleman in 1993. In essense, however, what developers of expected DNA computers need are computational theories and functional implementation procedures.

Formalization and modeling are used to construct the fundamentals of DNA computational systems. By means of automata, it is possible to formalize a kind of universal DNA computational model called EiCo, to explore its computational ability and its relationship with the Turing machine. In order to describe the corresponding characteristics, the necessary grammar representations and attributes with respect to the model theory (which is defined in terms of metamathematics) are adopted as one of the basic tools. Owing to the meta-evolution of emergence among the parallel architectures of evolution, the meta-evolution mechanism can be used to demonstrate the behavior of emergence at the genomic level in comparison with DNA manipulation in a tube environment by modern molecule biological techniques. This effort is aimed at verifying the DNA computational implementation for CBM in the form of a Darwin machine and Darwin chips.

Focusing on the NP completeness existing at the foundation of real world problems, which any information processing system has to face, DNA computation should step into the domain of complexity involving nonlinear dynamics such as chaos and symbolic dynamics and should establish certain logic forms for the cognitive modeling of brain activities. Several open problems remain in this field and deserve more attention. As an example of unveiling the nature of brain dynamics, some schemes based on KDD (Knowledge Discovery in Database) will be worked out to analyze the neurological and physiological mechanisms that cause people to suffer neural degenerative diseases such as Alzheimer's disease and Parkinson's disease in terms of the parallel hardware simulation of DNA computation.

DNA sequencing can provide the necessary means of verifying prediction results of unknown functions and structures by data mining processes. Data-driven and hypothesis-driven approaches can be integrated for greater efficiency and practicality of the underlying

system. Some significant evidence in molecular biology has shown that references are important in prediction works.
With respect to application in life science itself, our final target for DNA computing systems is to discover those genes that cause Alzheimer's disease but have not been known to humans till that time. The sample objects are expected to possibly be selected as some genomes in primitive life forms when experimental domains come to be concentrated into relative specific domains.

2 Meta-evolution of Emergence, Conditional Diversity, and Controllable Evolvability

In artificial life systems, the dynamical mechanism within the evolutionary processes is related to various kinds of nonlinear features. Complexity and diversity are two crucial factors. Behind them is the emergence mechanism. The origin of emergence can be explained by chaos and the direction of evolutionary processes can be analyzed by observing the tendency of complexity changes causing the characteristics of the dynamical mechanism in the underlying system.
Our research motivation is based on the idea that the meta-evolution of emergence is a crucial way to demonstrate the evolutionary dynamical mechanism at the genomic level, such that we define the concept --- genomic dynamics --- as the evolutionary dynamics of genomes.
The structure of our proposed model can be comprehended explicitly from its name --- the (l + m + n) model for the meta-evolution of emergence. Here l, m, and n take integers. The framework of this model is illustrated in Fig. 1.
The multi-dimensional mapping consists of coupling chaotic equations where homogenous and heterogeneous approaches can be selected according to the objects met in the domain concerned.
EC denotes the evolutionary algorithm with the general meaning and the remaining part described in Fig. 1 belongs to the meta-evolution of emergence. As an example, we can make the assumption that

$$l = 2,$$

$$m = 2,$$

$$n = 1$$

where the total dimension is 5 and an internal structure that corresponds to a certain subspace also exists within the 5-dimensional space. Then, we can assume that the mapping between l and m is adopted as the Gumowusky and Mira equation.

$$x_{i+1} = y_i + b\,(1 - 0.05\,y_i^2)\,y_i + f(x_i)$$

$$y_{i+1} = -x_i + f(x_{i+1})$$

where $f(x_n) = a\,x + 2\,(1-a)\,x^2 / (1+x^2)$.

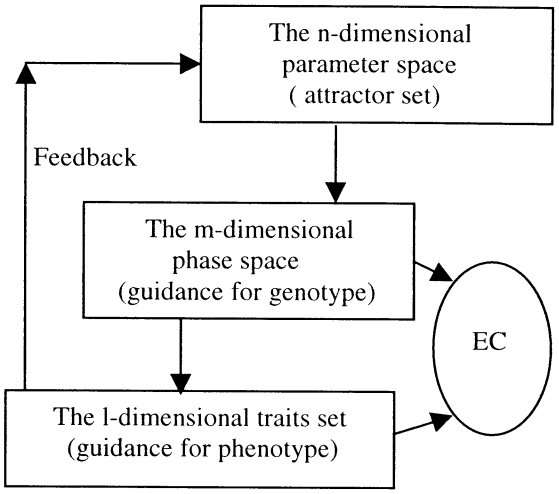


Fig. 1 Overview of a (l + m + n) model

The mapping from n to m is a Logistic equation.
As a result, after we estimate the trait based on the 10,000 sampling points, we can find that in the meta-evolution domain several areas are responsible for emergence. Figure 2 shows a typical case.

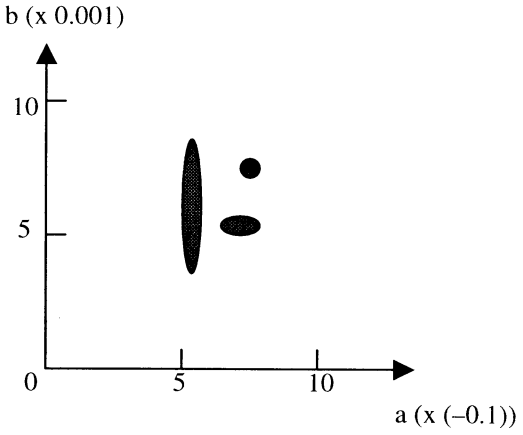


Fig. 2 An emergence area

Then, we can define the following two concepts --- conditional diversity and controllable evolvability.
Conditional diversity refers to the degree of occurrence of traits within certain areas. Controllable evolvability reflects the evolution ability with controlled tendencies owing to the underlying mechanism such as the meta-evolution of emergence. Briefly speaking, the former is a phenomenon and the latter is a feature. The meta-evolution mechanism for emergence connects them.
In the framework we built above, the computational power is expected from the complexity in dynamical systems.

3 Dynamical Symbolic Pattern Mining by a Tough Set

Focusing on the symbolic information from genomes and proteoms, pattern mining is constructed as a symbolic approach with an embedded dynamical mechanism.
Let U and V be the two sides of the boundaries covering the problem domain discussed. Then we can do the following.

U = the set that contains the objects with the minimum limitations,

V = the set that contains the objects with the maximum limitations.

A tough set is defined as a family of sets in which convex constraints and smoothing conditions are not necessary. Within the space derived from the corresponding evolutionary processes, only the approximation is the final target and efficiency is the key factor we attempt to obtain in the dynamical complex system rather than random selection.
As a concrete instance of the abstract algebraic systems we initiated [3], a tough set scheme for problem solving is proposed based on a "ring".

Definition 1 (Ring)
Let L be an algebraic system with two addition and multiplication operations; it is called a "ring" if it is satisfied with the following conditions:
(1) L can construct a changeable group for addition,
(2) L can construct a semi-group for multiplication,
(3) within L exists the operation that

$$a \$ (b + c) = a \$ b + a \$ c$$

where $ is an operator in L.
Let u be the unit element of L as an addition group. Then, the inverse element for addition v is –u, if w_l or w_r exists in L, and for any element y for any L

$$w_l y = y$$

or

$$y w_r = y$$

exists under the condition mentioned above; w_l is called the left unit element in L. In a similar way, we can define the right unit element in L.
Note that for any element z that exists in L, if it satisfies the features of the left and right elements, then it is called the unit element in L.
The corresponding algebraic system with the unit element, which satisfies the condition mentioned above, is called the ring (unitary ring).
With the algebraic representation and description of the ring, the relationship on the tough set can be defined and various types of evolving paths can be derived to handle different problems in specific cases.

4 Context-Sensible Grammar for DNA Computing

Induced by the dynamical evolution with the tough set, the grammar generated by the above-mentioned processes belong to the category of context-sensible grammars. The significance of DNA computational forms based on context-sensible grammars can be perceived at the implementation of corresponding functional automata, which are convenient and practical for hardware evolution (e.g., CBM).
For U and V, the measures can be denoted as m(u) and m(v). Within the range [m(v), m(u)], the production set obeys the following rule

$$\alpha_1 \Theta \alpha_2 \rightarrow \alpha_1 \Sigma \alpha_2$$

where α_1, α_2, Θ, Σ are assigned with the meanings that are required by the context-sensible grammar.
In our experiment, the symbol set is arranged as

S_g = { A, R, N, D, C, E, Q, G, H, I, L, K, M, F, P, S, T, W, Y, V}

where the elements within S_g are amino notations.
Given that the input is set into the proteome sequence

EN-DGVFMSVGYQIGQ-AVQQV-

with the molecular biological meaning of HopA in H. pylori[1], we get the parsing tree shown in Appendix [2].

5 Conclusion

DNA computing is a new and promising field in computer development, and its aim is to solve problems that existing computers are unable to solve. Our aim is to build a new kind of computer system based on the Super-Turing machine with a much stronger computational ability than that possible with current techniques --- equivalent to the Turing machine.
Besides computation mentioned above, logic is another important feature of computational machines, but Gödel's incompleteness constrains some functional factors when we try to construct formal systems. We are tying to overcome this problems with schemes that utilize the ancient oriental philosophy.
Beyond the limits of Gödel's incompleteness is another task in our DNA computing research.
At present, we are building a Genetyllis system based on a new kind of structure --- a hierarchical circular genome representation and its corresponding operations for genomes. Fortunately, with the help of our new model, we have found one genome reported no where else in the world and the creation of at least two protein sequences. The model has the potential for application in genome (bio)informatics [3,4].

In a word, the emergence mechanism acts as the fundamental factor within the DNA computer systems, and a kind of formal description --- context-sensible grammar is derived. Furthermore, a prototype of DNA computers is discussed in [5]

References

[1] Tomb J-F, White O, Kerlavage AR, et al. (1997), The complete genome sequence of the gastric pathogen Helicobacter pylori, Nature, 388: 539-547.

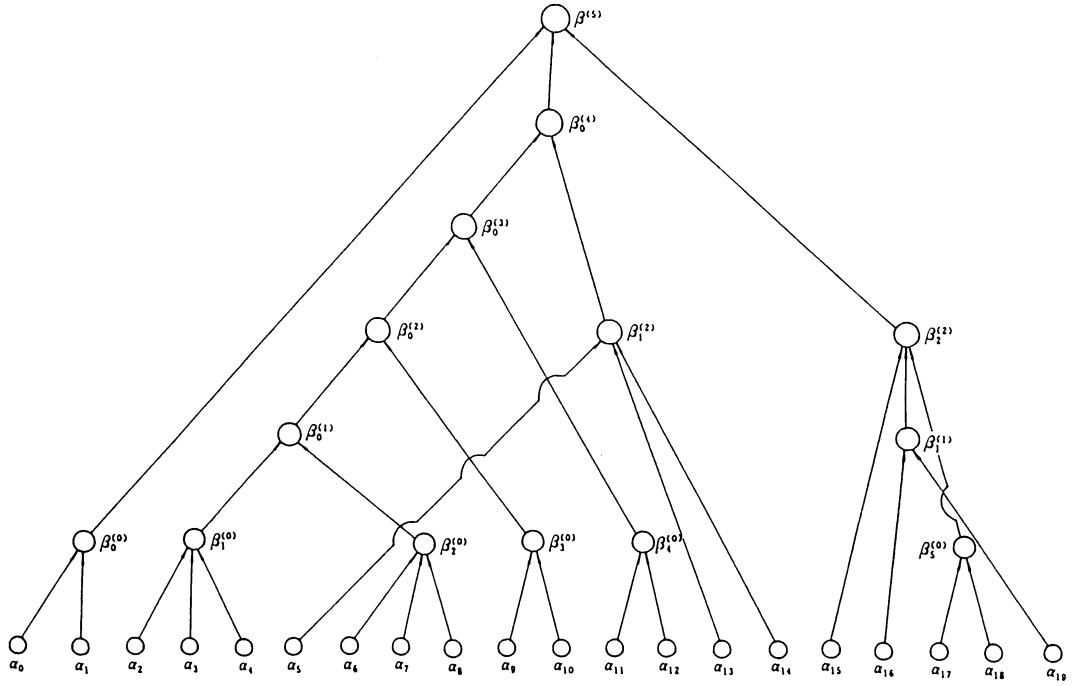
[2] Liu J-Q (1998), On pattern calssifier designing for N-terminal peptide sequence of pathogen H. pylori genome (in Chinese), Journal of Central South University of Technology, 29(6): 589-592.

[3] Liu J-Q, Shimohara K (1999), On emergence of evolutionary dynamics I: mathematical analysis, In: Proceedings of the 1999 Annual Conference of Japanese Neural Network Society, Sapporo, Hokkaido, Japan, September 20-22, 1999, pp.24-25.

[4] Liu J-Q, Shimohara K (1999), On emergence of evolutionary dynamics II: Demonstration in silicon, In: Proceedings of the 1999 Annual Conference of Japanese Neural Network Society, Sapporo, Hokkaido, Japan, September 20-22, 1999, pp.26-27.

[5] Liu J-Q, Shimohara K (2000), DNA computing by genomic dynamics II: --- a simulation wetware prototype of dynamical DNA computation. In: Sugisaka M (ed), Proceedings of the International Symposium on Artificial Life and Robotics (AROB 2000), Oita, Japan, Jan 26-28, 2000.

Appendix



An example of parsing tree with a context-sensible grammar for DNA computing

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

DNA Computing by Genomic Dynamics II: --- A Simulation Wetware Prototype of Dynamical DNA Computation

Jian-Qin LIU and Katsunori SHIMOHARA

ATR Human Information Processing Res. Labs., Evolutionary Systems Department,
Soraku-gun, Seika-cho, Hikaridai 2-2, Kyoto, 619-0288, JAPAN

E-mail: {jqliu, katsu}@hip.atr.co.jp

Abstract

Symbolic sequence representation is a crucial step in the molecular implementation of genomic DNA computers and it directly affects the information capacity of wetware architectures. Based on a multi-layered hierarchical structure, an artificial genome pool is generated and controlled. Within this pool, genome sequences are represented by a nonlinear dynamical structure with the same function for coding and decoding. We propose and implement a novel prototype to simulate a wetware architecture.

This architecture of DNA computing, which we construct to demonstrate the computational performance, consists four major units: (a) conversion for double string forms of genomes; (b) circular genome representation and generation; (c) dynamical genome arithmetic unit; and (d) evolutionary controller of coupled symbolic dynamical processes.

The original contribution of our work includes (1) a set of two symbols rather than the usage of four symbols; (2) infinite-length circular genome sequences with an infinite number of circles; (3) dynamical evolutionary computation with a genome structure (which has the potential for a stronger computational ability than the commonly-used universal computation form); and (4) procedural guidance by information from characteristics of real genome sequences such as alignment, clustering, and ORFs (Open Reading Frames). Prototype designing methods are discussed in comparison to generic features in E. coli and other species used frequently as references. Furthermore, several examples are demonstrated for quantitative description and measurement among the dynamical DNA computing paradigms and further implementation issues on evolution in vitro are discussed.

Keywords: DNA computer, bioinformatics, functional genomics, artificial life, nano-computing.

1 Introduction

In the underlying architecture of a DNA computer, some factors such as the sequences, operators and their corresponding dynamical operations, hierarchy and its coupled relationships, and application in genome (bio)informatics, should be taken into consideration when the corresponding wetware system is being simulated by a software approach. Evolution in vitro has similar features to artificial evolution. For example, features in controlling processes can influence the direction and affect the underlying evolutionary system. In the design unit, a sufficiently large sampled population size with a disturbing bias has shown that different trends exist from commonly-applied evolutionary algorithms.

ApEn can be used as a kind of macro measurement to describe the existing complexity. Manifold also provides a means of analysis for global optimization in the information space.

To develop a new kind of computer system based on genomic dynamics is one of our tasks at the moment for the new Millennium, the aim being to explore a mechanism that integrates life, information, and communication, and for biologically inspired communication techniques --- molecular computers.

To date, our work can be described as follows.

(1) We have originated and preliminarily developed a prototype simulator called Genetyllis for DNA computing. The prototype is constructed by a dynamical genome language and a super-Turing evolution machine --- our proposed EiCo (Evolution in corpore) model and related schemes.

(2) With the Genetyllis system, we have found a genome reported no where else in the world and it can produce at least two protein sequences with biochemical features.

(3) With the number system derived from Genetyllis, we have suggested a novel logic form --- the tough set --- which has a potential power beyond the limits of existing logic systems. The tough set is beneficial to the self-verification of signal systems through a configurable fine-grained architecture.

Our basis ideas are based on what we call the "Central Law" as Figure 1 shows. The evolutionary dynamics of genomes relates to our proposed concept, i.e., genomic dynamics, focuses which on analyzing the functions of the information we obtain from genomes and their applications. Its aim is to explore complexities in nature in terms of genomics. In the direction of increasing complexity, more novel functions emerge from the underlying system. Meanwhile, the emphasis for applications is on

structurally analyzing typical samples such as E. coli and H. pylori.
Upon studying genomic informatics further, we may find that common features exist both in natural evolution and artificial evolution. The proposed wetware architecture is related to both molecular biology and computer engineering. The formal language, Turing machine, and corresponding automata are factors that act as the necessary foundation of the computing system, even though the system is not a conventional one.
In order to implement a simulation of the digital evolution in silicon (hardware), parallel distributed processing schemes such as VPM (Virtual Parallel Machine) and networks can contribute to the architecture to a certain degree. Of course, our final target is artificial life in vitro with molecular implementation towards nano-computers where some revisits to the concepts of "DNA Computing/Computers" are inevitable.

2 Quantitative Description in the Genetyllis System

2.1 Symbolic Representation for a Genome

Consider a finite symbol set G = $\{g_1, g_2\}$ where g_1= A/G and g_2 =T/C. We can define the set of all of the finite sequences over G as G^+ and the set of all of the infinite sequences over G as G*.
Each genome sequence $L = l_0 l_1 \ldots l_n \in G^*$.

2.2 Activating the Index in Structural Units

Besides being a controlling mechanism, the index i in reference to the lth circle is given by

$$i(l+1) = f(i(l), \Psi)$$

where f (.) is set into a numeric chaotic dynamical function and
Ψ is a set of the parameters concerned.

2.3 Symbol Generation in a Dynamical Way

The symbol S is generated in a symbolic way as follows

$$S(l+1) = h(f(g(S(l), \Theta_g), \Theta_f), \Theta_h))$$

where

Θ_g, Θ_f ,and Θ_h ---- the sets of the parameters corresponding to g, f ,and h;
g(.) ---- the nonlinear mapping from a symbolic variable to a numeric one;
h(.) ---- the nonlinear mapping from a numeric variable to a symbolic one.

The features in the structure concerned with dynamical genome representation include generalization, characterization, and homeostasis in view of the complexity.
As shown in Fig. 2, the architecture of Genetyllis can be described as consisting of the following three major parts:

(1) The Left Unit:

This structure consists of circles with the same center with infinite lengths (the ideal case). Each circle can be regarded as a tape in which the symbol S is generated (written) by the OP in the middle unit (the upper layer compared to the former). We can assume that the initial state of the circular tape is empty or it can be replaced in an iterated way when we do not consider the previous state (i.e., it can not influence the current state or any future states written on it). Note that the final genome we want is generated as a straight line.

(2) The Middle Unit and Related Right Unit:

To induce the left unit, the middle unit is arranged as a kind of symbolic dynamics with deterministic chaos embedded. The index i in Si has no relationship with the index j in Sj. The occurrence of i is carried out by OP and that of j is done by itself.
The difference is made to distinguish the inner and outer circles as two symbol sets

$$S_{outer} = \{T, C\}$$

and

$$S_{inner} = \{A, G\}$$

This approach is different from the direct use of a set of four symbols ({A, T, C, G}) as reported by others.
Furthermore, the index j of OP that occurs both in the inner circle and the outer one is determined by the right unit SD, which also works in a similar way.

(3) General Form:

Within the multi-layered hierarchical structure mentioned above, each dynamical unit is arranged in a coupled "chain". In this chain, they can be set into one of the following structures: homogeneous, heterogeneous, or hybrid (integrated) structure.
In addition, the chaotic function can be set as a simple form (e.g., logistic function) or some other higher-dimensional form with a coupled mechanism (e.g., coupled Gumowusky or Mira function embedded with a logistic function).
The significance lies in the following two points:

(a) Any Feature in General

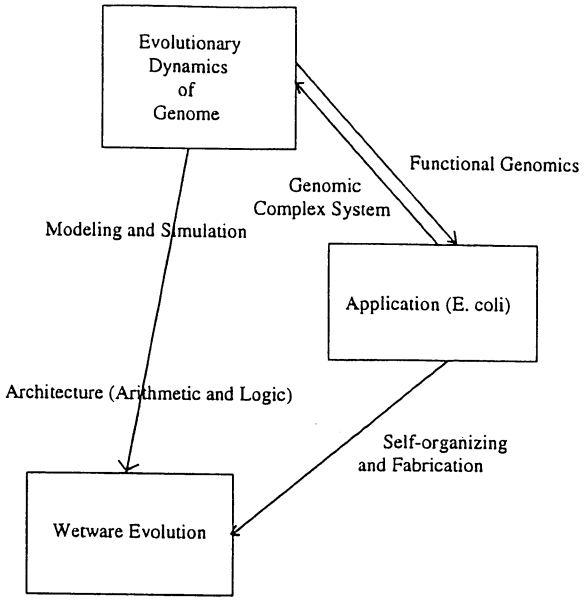


Fig. 1 The "Central Law" in the view of ALife

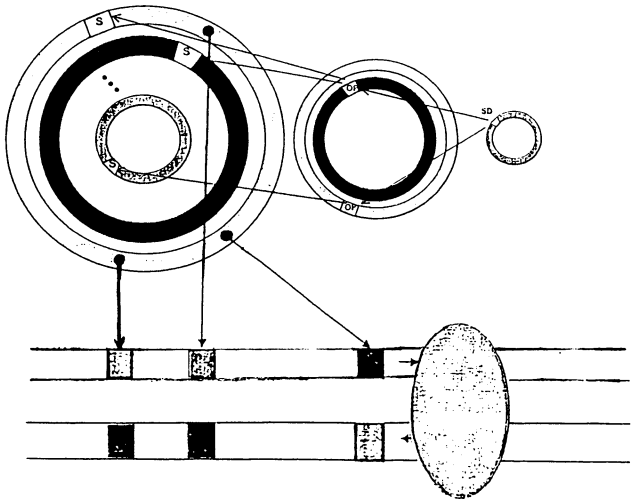


Fig. 2 The structure of the Genetyllis system

Note that some notations are given as follows.

S --- one symbol in a genome

OP --- dynamical operation

SD --- symbolic dynamical controller

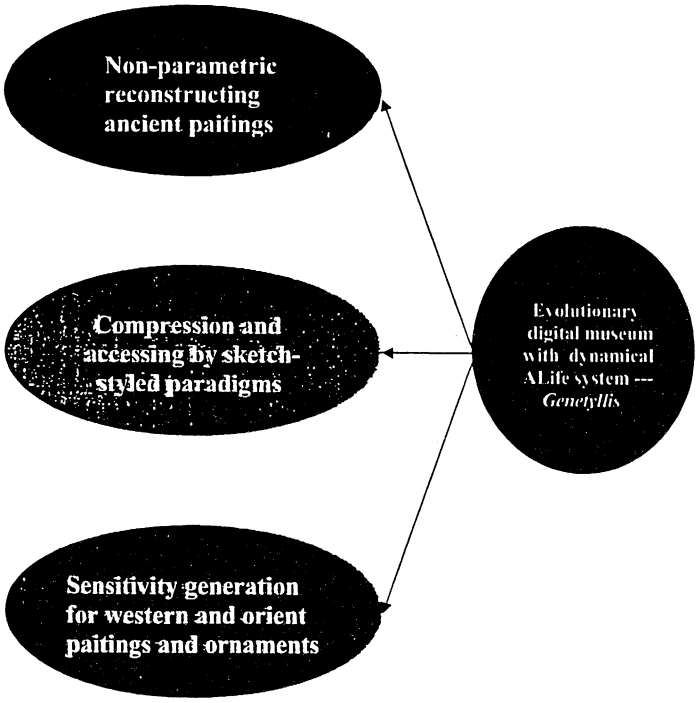


Fig. 3 **The openness in *Genetyllis* to visual digital museum**



The recognition result (probability) of boy or girl

(a) (b) (c) (d)

Boy, Similar to boy, Similar to girl, Girl

It is for the same child who actually is a girl but the hints such as hair type, clothes, environment and etc. are changed in different situations.

Fig. 4 Sample 1

In the space with the representation of the vectors combined from all of the variables and parameters in the whole system, the features in the path of the states to transverse (travel) can reflect the various types of changes in the circular sequences in the landscape. Note that the dynamical generation scheme discussed here contrasts the random selection approach reported by others.

(b) From a View of Bio-informatics

The scheme contrasts the commonly-used machine learning paradigm in bio-informatics and is not limited to the concept of "evolutionary dynamics" in ALife.

3 Functional Extensions in Genetyllis

Genetyllis application lies obviously in encryption for network media in cyberspace owing to its strong ability in evolvability and complexity, and sensitivity (kansei) technologies for human-to-human and human-to-machine communications (e.g., visual pattern mining and discovery for object segmentation and recognition). The openness of Genetyllis to a visual digital museum is given in Fig. 3.

A clear-cut prototype which we are now constructing has its focus on the following three directions in which the two streams of art and engineering are integrated within an ALife system to support "kansei" communications:

(1) The embedding of psychological functions into visual agents:

Besides evidence showing that the attractiveness of faces gives different tendencies in males and females on human impressions, a psychological influence also exists for the recognition of males and females and causes certain tendencies as we show in sample 1. Naturally, Genetyllis has an emergent mechanism to model the crucial mechanism that leads to the construction of self-perception in the artificial life system to visually communicate with humans of different favors.

(2) The construction of aesthetic behaviors in artificial life:

Sketches focus on key factors that maintain enough information about the creation processes of visual art. Accordingly, we apply them as a way to abridge the gap between the concepts of art and the pixels of media forms.

(3) An ALife system towards psychological category application and evolving circumstances:

The emergence among the agents is oriented towards ad hoc behaviors for "kansei" such as the appreciation of visual artworks and active interaction in communications within digital museums.

To demonstrate our proposed ideas mentioned above, we have carried out a psychological experiment on subject tendencies for male/female face expressions from photos (Cf. Fig. 4 and Fig. 5). We have also attempted to extract "kansei" information from visual artworks (Cf. Fig. 6) in order to abridge the gap between art and engineering. With our progress so far in the interaction between psychological observations and communication technologies, we have been trying to create next-generation communication systems, which will make non-verbal communications between humans and machines much more efficient than with current techniques.



The different factors for hints with different combination

Fig. 5 Sample 2





© 1998, Liu Qixing © 1999, Jian-Qin LIU

Fig. 6 The artworks (painting: left, sketch: right)

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Personal Adaptation of Human Touching Behavior Recognizer using Pressure Sensors

Kazuhiko Shinozawa, Futoshi Naya and Junji Yamato

NTT Communication Science Labs.
2-4 Hikaridai, Soraku-gun, Seika-cho Kyoto 619-0237 Japan

Abstract

This paper presents a user adaptation method for the classification system of human touching behavior we have developed with a high-speed pressure sensor array.

We have studied how human touching behaviors can be classified by using pressure distribution because "touching" each other is an essential communication channel for human-robot interaction. Three features of pressure patterns,-sum of absolute values of each sensor element, spatial distribution, and temporal differences- were proven to be sufficient for classifying five touching patterns: "Slap," "Pat," "Scratch," "Stroke" and "Tickle".

We introduced an adaptation mechanism for each user because a general classifier cannot focus on detailed personal behavior patterns and preferences. Furthermore,a general classifier has static response. This means that it may infinitely repeat miss-classification of the same input. This is tedious in human-robot interaction. Therefore, the adaptation to personal behavior patterns and their changes based on an interactive learning mechanism is important for robots that interact with humans. The adaptation mechanism is a type of on-line learning based on user's "punishment" behavior, which the classifier has to recognize by itself. We adopted the k-Nearest Neighbor method for our classifier system, and through a punishment signal, the previous data item is used as negative training data for user adaptation. The new classifier system with its user adaptation feature demonstrated a quick response and better performance than general static classifier.

We are developing a robot that has multi-sensor sheets on its body and head. This algorithm will be implemented on the actual robot and will be demonstrated in an interactive situation in the real world.

1 Introduction

Haptic information is expected to play a great role in enriching communication between humans and robots in the real world. In interactions between humans and robots, distinguishing haptic patterns is very important. In particular, this is useful for helping a robot to recognize human touching behaviors and also for having the robot express its internal states, or artificial emotions, by physically touching humans. We are currently promoting *the Communication Robot project.* The goals of this project are to develop a robot that can naturally interact with humans through touch and to measure human emotions through the physical interaction between humans and robots.

There has been much research on haptic devices designed to develop tactile sensors for robot hands and fingertips [1, 5]. These sensors have mainly been designed to detect contact between a robot's hands and the target objects to be grasped or manipulated. However, our goal is to have a robot recognize *human touch*, which means we need to develop tactile sensors that have a relatively wide sensing area and the flexibility to be used to cover the entire body of the robot.

Few sensors have been developed for measuring human touching behavior. Inaba *et al.*[2] have developed a tactile sensor suit that uses electric conductive fabric to cover the entire body of a humanoid robot. This sensor suit is flexible and has 192 sensing regions, which are binary switches. The sensor signals are gathered and superimposed on a visual image of the robot so that the sampling ratio of the sensor data is equal to that of the video frame rate (30 Hz). The study presented sensing pattern examples of a human touching the robot's arms and the robot holding a ball.

In another study, Shibata *et al.*[4] proposed a tactile sensor for a pet-like robot. This tactile sensor uses a semiconductor pressure sensor that is connected to a balloon covered by synthetic skin. This sensor is

designed to be comfortable for humans. The study presented the sensing patterns of human touching behaviors such as 'patting' and 'stroking,' but did not give much detail on the recognition scheme for these behaviors.

In order for a robot to recognize more sensitive patterns of human touch (e.g. a tickle, scratch), haptic devices are needed to measure touching pressure at a high sampling rate and at high spatial and pressure resolutions. We pointed out that five special human touching behaviors can be classified in off-line by using a gridded pressure sensitive sheet and the k-Nearest Neighbor method with three features of pressure patterns[3].

When we imagine the interaction between robots and humans, if the classifier system usually misclassifies the human touch behavior and never changes the results, the user would naturally frustrated. In addition, if the user must teach the right result and if its method is much different from natural touch behavior to the robot, the user may become even less inclined to interact with it.

Thus, we need an on-line adaptation method for this classifier system without special touching behavior patterns.

In the next section, we describe our new on-line adaptation method.

2 On-line Adaptation Method

We focus on four human touching behaviors; "Slap," "Scratch," "Stroke," and "Tickle". We assume that users always interact with the robot, through the above four touching behaviors and focus on the "Slap" behavior. When users do the "Slap" behavior, they scold the robot and show disapproval to the robot, so it is a natural situation for the user to do the "Slap" behavior when the classifier system of the robot misclassifies.

We use the "Slap" behavior as punishment of the previous classification in the classifier system and consider a new on-line adaptation method for four touching behaviors with this punishment. Thus, we assume typical users always slap the robot whenever its classifier system misclassifies.

2.1 Features of patterns

We use the following four features of pressure patterns, which were used before[3].These features are normalized by each distribution.

1. The normalized total load of a two-dimensional pressure pattern at time t.
2. The normalized total contact area of a two-dimensional pressure pattern at time t.
3. The normalized temporal difference of the total load between time t and $t-1$.
4. The normalized temporal difference of the total area between time t and $t-1$.

2.2 Improved k-NN method

We adopt the k-Nearest Neighbor (k-NN) method for classifying touch in four-dimensional feature space as described above. The conventional k-NN method determines the reliable category with a majority of nearer training data points. In other words, the k-NN method has only positive training data points(ψ^+). We introduce negative training data points (ψ^-) to the k-NN method and give the k-NN method the ability to deny the category that the negative training data point belongs to. These modifications give the classifier system user adaptation

Our improved k-NN algorithm is as followings;

1. Select k nearest training data points (prototype) $P_i^t (1 \leq i \leq k, P_i^t \in \psi, \psi = \psi^+ \cup \psi^-)$ around the feature vector X^t of new touch behavior at time t.
2. For all i , calculate the $V(c)$ for the category c.
$$V(c) = \begin{cases} V(c)+1 & if \quad P_i^t \in \psi_c \cap \psi^+ \\ V(c)-\alpha & if \quad P_i^t \in \psi_c \cap \psi^- \\ & \alpha = f(||X^t - P_i^t||) \end{cases}$$
Function f and ψ_c are monotone decreasing function and training data points belonging to the category c, respectively.
3. Choose the category $C^t = \arg\max_c V(c)$ for X^t.
4. If the C^t is "Slap" behavior, register the X^{t-1} as ψ^-, and if the C^{t-1} is "Slap" behavior, remove the X^{t-2} from ψ^- that registered it at time t-1.

Here, it is assumed that the previous classification is a failure when the robot is slapped, so if the robot is slapped, the data point used for the previous classification is registered as a negative training data point.

3 Examinations

To evaluate our method, we conducted a simulation experiment of the classification. We evaluated our method by using the data obtained in previous experiments[3].

3.1 Conditions

One test subject is chosen from the entire group of ten, and the adaptation evaluation experiment starts the classification by using the data of the other nine testees. Moreover, the user's data are divided into B data group slapped and another A data group beforehand.

It is assumed that the user generates the touch behavior from the category included in A group at random and generates behavior from B group at random for misclassification. We used the data included in A group with the classifier at random in the simulation. If the classifier is mistaken, we used the data included in B group at random, and if the data is classified as "Slap" behavior, the previous data is registered as negative training data. Moreover, classification uses the improved k-NN method as described above.

Because this result depends on the behavior sequence, we evaluated the method by using the average of the results produced with 100 different behavior sequences. In this experiment, we set the value of k to five.

3.2 Experimental Results

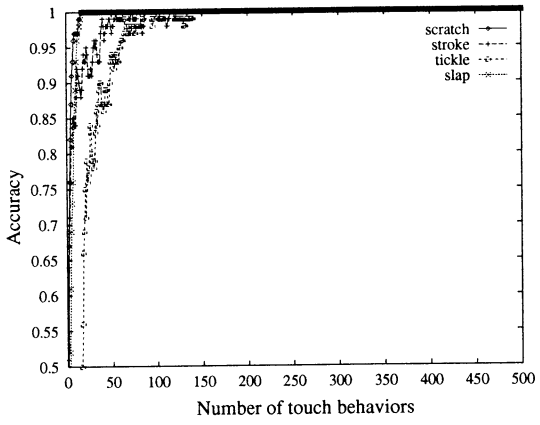


Figure 1: Average correct ratio for Subject 1

Figure 1 shows the results of a user. The X axis in the figure shows the number of behaviors taken by the user , and the Y axis shows the average correct answer ratio over 100 variant touch behavior sequences.

The figure also shows that the average ratio of correct classification approaches one and that the adjustment advances as the frequency of the user's touch behavior increases. The nine results of 10 test subjects showed that the classification with our method could successfully adapt to the user as well as to subject 1.

However, Fig. 2 shows that the correct classification ratios of some touch behaviors did not become one. We

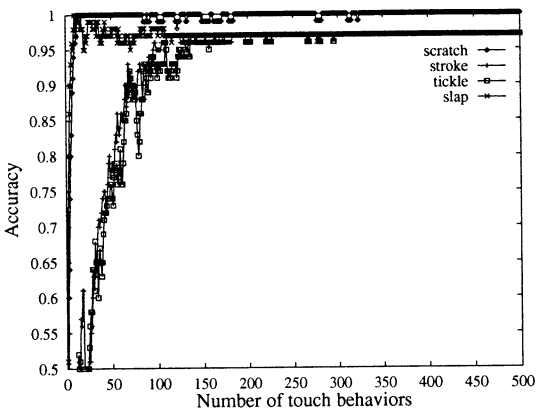


Figure 2: Average correct ratio for Subject 2

used the average ratio of correct classification as the evaluation of the method, so this result means there are some failures which the adaptation could not be done at all for the some behavior sequences.

3.3 Analysis of Error

Our method adapts to the user only by the "Slap" behavior, so the correct classification ratio of "Slap" behavior before adaptation is very important.

	Scratch	Stroke	Tickle	Slap
Scratch	1.00	0.00	0.00	0.00
Stroke	0.500	0.333	0.00	0.167
Tickle	0.00	0.625	0.375	0.00
Slap	0.00	0.00	0.00	1.00

Table 1: Classification ratio before Subject 2 adaptation.

	Scratch	Stroke	Tickle	Slap
Scratch	0.999	0.001	0.000	0.000
Stroke	0.003	0.992	0.000	0.005
Tickle	0.004	0.004	0.991	0.001
Slap	0.008	0.032	0.000	0.960

Table 2: Average classification ratio after Subject 2 adaptation.

Table 1 shows the classification ratio results of Subject 2 before adaptation. The column and the row indicate the classification results and the correct behavior, respectively. There are a few misclassification results where the system classifies the stroke behavior as the slap behavior. This misclassification causes the wrong learning, which denies the previous result although it is correct. Its influence may spread to the classifica-

tion results of all behaviors according to a sequence of the user's behaviors. In the worst case, our method infinitely repeats misclassifying the stroke behavior as the slap behavior and keeps the system memorizing the wrong negative training data. Table 2 shows the average ratio of the classification of user touching behavior after Subject 2 adaptation. The diagonal of the matrix means correct classification ratios. They are better than before the adaptation, but the ratios of misclassification do not become zero for the reason mentioned above.

The data of "Slap" behavior might be classified as other behaviors, contrary to the above-mentioned case. This does not cause worse results than described above although the punishment from the user might not be able to be recognized. In addition, there is no bad effect to the future classification results because no learning mechanism is performed.

The misclassification often originates in slapped strength. There is an expectation that the learning of the user will cause stronger beating when the identification result is presented to the user by actually using this method. Moreover, it is thought that this problem can be avoided by instructing the user beforehand to slap strongly when the identification result is bad.

For Subject 1, there is no misclassification of the "Slap" behavior, as Table 3 shows, and the adaptation makes good progress. Furthermore, the ratio of correct classification becomes one, as Table 4 shows.

	Scratch	Stroke	Tickle	Slap
Scratch	1.00	0.00	0.00	0.00
Stroke	0.00	0.91	0.09	0.00
Tickle	0.00	0.625	0.375	0.00
Slap	0.00	0.00	0.00	1.00

Table 3: Classification ratio before Subject 1 adaptation.

	Scratch	Stroke	Tickle	Slap
Scratch	1.000	0.000	0.000	0.000
Stroke	0.000	1.000	0.000	0.000
Tickle	0.000	0.000	1.000	0.000
Slap	0.000	0.000	0.000	1.000

Table 4: Average classification ratio after Subject 1 adaptation.

4 Summary

We proposed a new on-line adaptation method for user touching behavior. Our method can adapt the classifier to the user without forcing a special method on the user. Therefore the user can interact with the robot naturally using our method.

The method is based on modification for the "Slap" behavior, which play the punishment role, therefore, the accuracy of classification for "Slap" behavior is very important for correct adaptation. Misclassification of other behaviors as a "Slap" sometimes has a bad influence on the on-line adaptation, while the opposite case, misclassification of "Slap" as other behaviors, does not cause serious problems.

4.1 Future Work

We will improve our method so that it classifies "Slap" behavior correctly. Our method will be implemented on a robot and, experimentation on on-line adaptation will be conducted by using the actual robot. We will investigate whether the learning of the user and the adaptation mechanism can work together.

References

[1] E. Cheung and V. Lumelsky. Development of sensitive skin for a 3d robot arm operating in an uncertain environment. In *Proceedings of the IEEE International Conference on Robots and Automation (ICRA'89)*, pages 1056–1061, 1989.

[2] M. Inaba, Y. Hoshino, K. Nagasaka, T. Ninomiya, S.Kagami, and H. Inoue. A full-body tactile sensor suit using electrically conductive fabric and strings. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'96)*, pages 450–457, 1996.

[3] F. Naya, J. Yamato, and K. Shinozawa. Recognizing human touching behaviors using a haptic interface for a pet-robot. In *Proceedings of the 1999 IEEE International Conference on Systems, Man, and Cybernetics (SMC'99)*, pages II–1030–II–1034, 1999.

[4] T. Shibata, M. Yoshida, and J. Yamato. Artificial emotional creature for human-machine interaction. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC'97)*, pages 2269–2274, 1997.

[5] J. Son, E. A. Monteverde, and R. D. Howe. A tactile sensor for localizing transient events in manipulation. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation (ICRA'94)*, pages 471–476, 1994.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A Conceptual Framework on Geino Organization Model for Artificial Geino Performers with Lives

Yoji Kawamura

Faculty of Economics, Osaka University of Economics and Law
6-10 Gakuonji, Yao-shi, Osaka, 581-8511, Japan
E-mail: kawamura@keiho-u.ac.jp

Takashi Ogata

Department of Computer Science and Media Engineering, Faculty of Engineering, Yamanashi University
4-3-11 Takeda, Kofu-shi, Yamanashi, 400-0016, Japan
E-mail: ogata@esi.yamanashi.ac.jp

Abstract

We have the plan to create a new integrated system of narrative technology, contents and business. We have formulated a series of research program for studying the social and folk phenomena from the narrative point of view. We are trying to model the information system of Japanese-Geino (performing art, entertainment, show-business, etc.) organizations and services which produce and introduce the narrative software.

At first, this paper shows an outline model of Japanese-Geino organization including the interaction between imaginary and substantial narrative, secondary explains a case analysis of Japanese-Geino performer macro history, and finally mentions a narrative macro strategy of artificial Japanese-Geino performers with lives.

According to the Japanese-Geino performer (Japanese idol) example, we can analyze that a typical form of the Geino performer macro history is "Idol image creation -> Image share by multiple media development -> Outside-environment change or scandal -> Drop or change in image." We can obtain a meta-history for the modeling of the Geino performer's life. And, we can model a narrative macro strategy cycle of the Geino producer for management of Geino performer. This strategy cycle gives a narrative macro strategy of artificial Geino performers with lives.

Keyword : Geino Organization Model, Artificial Geino Performer, Narrative Strategy, Substantial Narrative, Imaginary Narrative

I. Introduction

We have the plan to create a new integrated system of narrative technology, contents and business. We have formulated a series of research program for studying the social and folk phenomena from the narrative point of view. We are trying to model the information system of Japanese-Geino (performing art, entertainment, show-business, etc., we say "Geino" below.) organizations and services which produce and introduce the narrative software[1][2][3]. The reasons why we try to research Geino phenomenon are below.

(1) Social Background

The contents of subcultures (such as comics, amusement, Geino, etc.) have global marketing power as Japanese original culture in recent years, and they are expected to become the future industry supporting Japan (for example, the latest U.S. Pocket-Monsters boom). And, in the marketing domain, the research on service (Geino is included) is recognized to be important field in recent years [4][5].

(2) Scientific Background

It is new trial to grasp the complicated cultural Geino phenomenon not in the literary, historical and folkloric method but in the system theory method. As Geino phenomenon include not only the substantial narrative but also the imaginary narrative, it is a synthetic subject of narrative system research.

(3)Technical background

Geino system is a new technology field to use computer science (such as artificial intelligence, artificial life, etc.) and new software (such as multimedia, thinking/creation support, etc.).

In recent years, several Geino producers are trying to create virtual idol (ex. DK-96, Yuki Terai[6]) in Japan. These trials are mainly the technical approach by advanced computer graphics technology. But the response of the receiver is not so good. According to the comment of the receiver, "virtual idol have not her life event (or episode), so we cannot use imagination." We think that this comment shows the important factor of the artificial Geino performer, so recognize that the essence of Geino (especially Japanese idol) system is the interaction between substantial narrative by the Geino producer and imaginary narrative by the receiver, and then try to model Geino organization by making clear the interaction between substantial and imaginary narrative.

At first, this paper shows an outline model of Geino organization including the interaction between substantial and imaginary narrative, secondary explains a case analysis of Geino performer macro history, and finally mentions a narrative macro strategy of artificial Geino performers with lives (as an interaction rule between substantial and imaginary narrative).

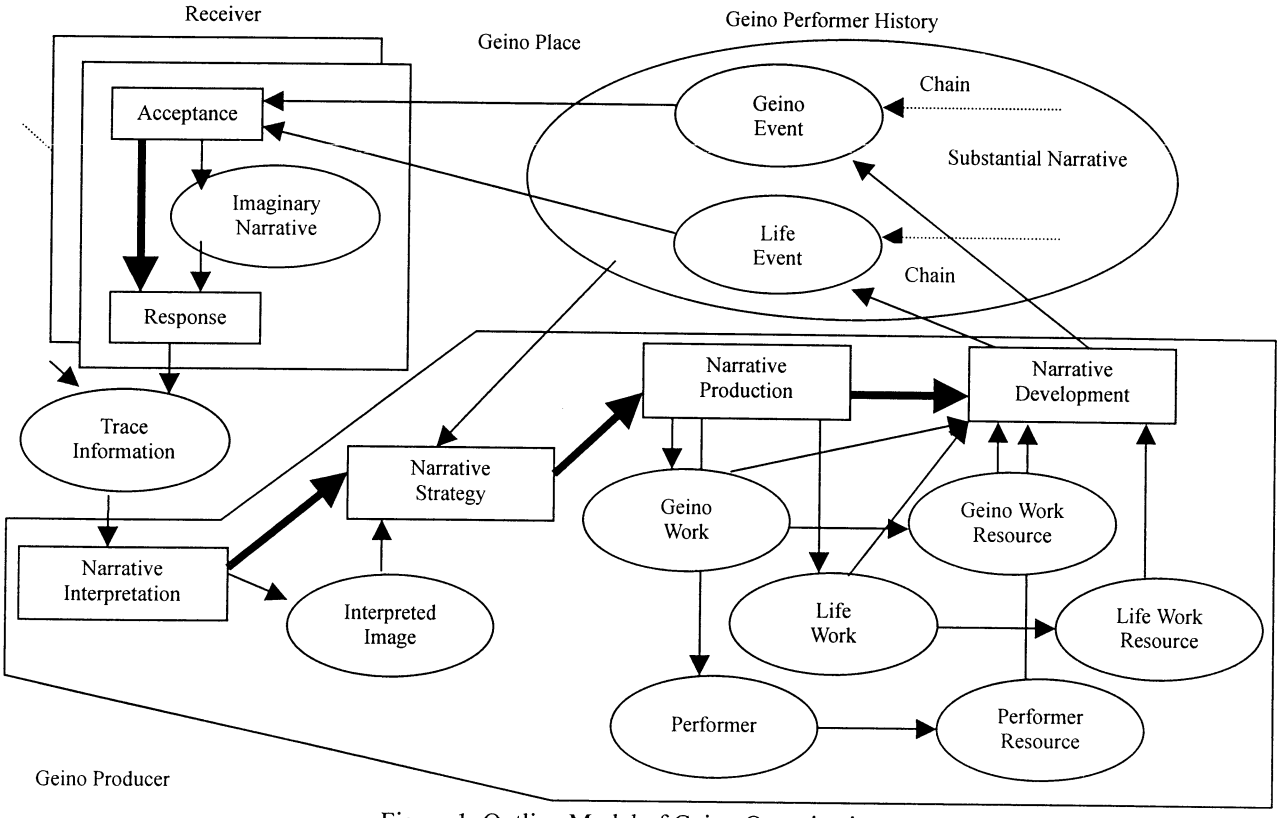


Figure 1. Outline Model of Geino Organization

II. Outline Model of Geino Organization

We show an outline model of Geino organization in Figure 1. Geino organization model is a mechanism that not only develops new substantial narrative (Geino and life event of performer) to receiver, but also facilitate imaginary narrative which receiver create. And, Geino organization model is a system which controls production and development of substantial narrative (Geino work) by interpreting imaginary narrative of individual and mass level.

Geino organization consists of various kinds of narrative subject (Geino producer, performer, other staff, receiver, etc.), narrative communication link (producer-receiver, performer-receiver, receiver-receiver, etc.) and narrative contents (Geino work, Geino event, Geino event chain, performer life event, performer life event chain, etc.). Based on its complex and multiple structure, Geino organization can create various kinds of substantial and imaginary narrative.

1. Geino Producer Mechanism

Geino producer mechanism consists of narrative interpretation mechanism, narrative strategy mechanism, narrative production mechanism, and narrative development mechanism.

(1) Narrative Interpretation Mechanism

A mechanism which generates the interpreted imaginary narrative from the trace information that shows the receiver's response to the Geino and life event.

(2) Narrative Strategy Mechanism

A mechanism which plans the micro and macro strategy for production and development of substantial narrative on the basis of the interpreted imaginary narrative and the Geino performer history. Micro strategy is a plan of the following work production and development, and macro strategy is a mutual systematization strategy between the interpreted imaginary narrative and the Geino performer history.

(3)Narrative Production Mechanism

A mechanism which conceives the Geino work and the Geino performer's life work (scandal is characteristic) and creates the scenario.

(4)Narrative Development Mechanism

A mechanism which determines expression media, casts, etc., designs work expression at Geino place, and develops the event concerning the Geino and life work.

2. Receiver Mechanism

Receiver mechanism consists of Geino place, acceptance mechanism, and response mechanism.

(1) Geino Place

A concrete situation that expression of the Geino or life work is developed.

(2) Acceptance Mechanism

A mechanism which receives the Geino and life event through a certain media.

(3)Response mechanism

A mechanism which expresses the trace (systematic or fragmentary) information about the response to the Geino or life event. This includes not only a mechanism within a brain but also a social matter.

III. Case Analysis of Geino Performer History

Based on an outline model of Geino organization, we investigated the phenomenon about several Geino performers (Japanese idol) and arranged the Geino and life event chain [2].

According to the Geino performer (Japanese idol) example, we can analyze that a typical form of the Geino performer macro history is "Idol image creation -> Image share by multiple media development -> Outside-environment change or scandal -> Drop or change in image". We can obtain a meta-history for the modeling of the Geino performer's life.

(1) Media exposure is performed as a supporting player of two or more works at the time of a debut.
(2) If notability goes up by a certain image (concept, such as a beauty boy, a pure group, etc.), popularity will be raised by developing various media (ex. record, drama, movie, concert, CM, magazine, etc.) on the basis of its image.
(3) Outside-environment change (ex. other Geino genre boom, other rival idol popularity, etc.) or scandal (ex. sweetheart disturbance, disappointed love, etc.) have occurred at grade for several years - ten years.
(4) Popularity of one performer is falling quickly or gradually by outside-environment change or scandal. On the other hand, although outside-environment change and scandal occur, the other performer can change their image (ex. maturity singer appeal, marriage with another performer, etc.) by taking them in as their life event, and success to get the impelling force of notability maintenance as a result.

IV. Narrative Macro Strategy of Artificial Geino Performer

Based on an investigation and analysis of current Geino organizations[3] and a result of III., we recognize that Geino performer can be characterized in a life cycle (Figure 2) from the viewpoint of the narrative expression technology's capability (immature, mature) of the Geino performer and the image creation power (individual level, mass level) of the receiver. We can model a narrative macro strategy and countermeasure cycle of the Geino producer for management of Geino performer. This strategy cycle gives a narrative macro strategy of artificial Geino performers with lives as an interaction rule between substantial and imaginary narrative.

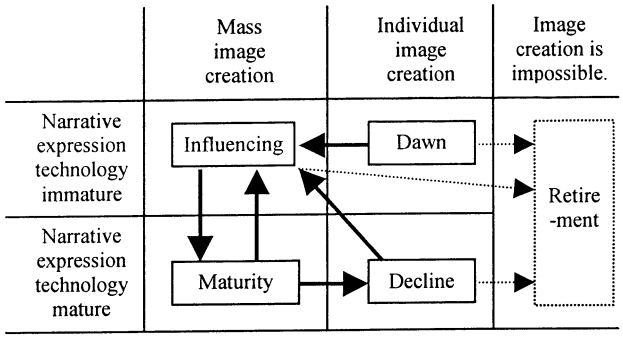


Figure 2. Geino performer's life cycle

1.Dawn : Image creation strategy

When the Geino performer's expression technology is immature, the Geino producer investigates the receiver's participation possibility by making the receiver's imaginary narrative creation, on the basis of the fragmentary information (such as profile, looks, voice, etc.). The following countermeasures of Geino performer image creation strategy are taken in many cases.

(1) If the Geino performer's expression technology is immature, the Geino producer develops the Geino event (S : a function of Geino performer is supporting player or opening performance) on the basis of the Geino performer's profile.
(2) If the receiver's response to the Geino event (S) is good, the Geino producer interprets the Geino performer's imaginary narrative from the response, and shifts to (1) of influencing time.
(3) If the receiver's response to the Geino event (S) is bad, the Geino producer develops the new Geino event (S) of the different Geino performer's imaginary narrative, or develops the new Geino event (S) suitable for the receiver's imaginary narrative, then returns to (2).
(4) If the receiver's response to the Geino event (S) is always bad, the Geino producer resigns the aggressive sales of the Geino performer.

2. Influencing Time : Image share strategy

The Geino producer sets the Geino performer's concept which the receiver tends to accept, develops the Geino event based on its concept for getting many receiver's participation by making image share and amplify, and brings up the Geino performer's expression technology in the meantime. The following countermeasures of Geino performer image share strategy are taken in many cases.

(1) The Geino Producer develops the new Geino event (M : a function of Geino performer is main player or final performance) on the basis of the Geino performer's interpreted imaginary narrative.
(2) If the receiver's response to the Geino event (M) is good, the Geino producer develops the Geino event (M) which utilized the new Geino resource (Geino work, Geino expression media and Geino place) on the basis of the Geino performer's interpreted imaginary narrative, and repeats this phase.
(3) If the receiver's response to the Geino event (M) is bad, the Geino producer develops the Geino event (S) which utilized the Geino resource on the basis of the Geino performer's interpreted imaginary narrative. If the response of the receiver to the Geino event (S) is good, the Geino producer re-develops the Geino event (S) which utilized its Geino resource, and repeats this phase.
(4) If the Geino performer's narrative expression technology is mature, the Geino producer shifts to (1) of maturity time.
(5) If the receiver's response to the Geino event (M or S) is always bad, the Geino producer resigns the influencing sales of the Geino performer.

3. Maturity Time : Image share or creation strategy

The Geino producer produces and develops the Geino event which employed maturity of the Geino performer's expression technology for getting the receiver's participation by making image share, and shifts the receiver's imaginary narrative to an image suitable for expression technology of the Geino performer gently. The following countermeasures of Geino performer image share (or creation) strategy are taken in many cases.

(1) If the Geino performer's narrative expression technology is mature, the Geino producer develops the Geino event (M or S) on the basis of the Geino performer's narrative expression technology.
(2) If the receiver's response to the Geino event (M or S) is good, the Geino producer develops the Geino event (M or S) which utilized the new Geino resource on the basis of the Geino performer's narrative expression technology. the Geino producer interprets the Geino performer's imaginary narrative from the response, and returns to (1) of influencing time.
(3) If the receiver's response to the Geino event (M or S) is bad, the Geino producer develops the Geino event (M or S) which utilized the different Geino resource on the basis of the Geino performer's narrative expression technology, investigates a set of the Geino resource which was suitable for the Geino performer's narrative expression technology from the response, re-develops the Geino event (M or S) which utilized its Geino resource, and repeats this phase.
(4) If the receiver's response to the Geino event (M or S) is always bad, the Geino producer shifts to (1) of decline time.

4. Decline Time : Image re-creation strategy

Although the Geino performer's expression technology is hackneyed, the Geino producer investigates the receiver's re-participation possibility by making the receiver's imaginary narrative re-creation, on the basis of the Geino performer's private life, occurrence incident, etc. When the imaginary narrative which the receiver tends to accept is able to be interpreted, the Geino producer shifts to influencing time. The following countermeasures of Geino performer image creation strategy are taken in many cases.

(1) If the Geino performer's narrative expression technology is hackneyed, the Geino producer develops the Geino event (S) and the Geino performer's life event on the basis of environmental cue (happening).
(2) If the receiver's response to the Geino event (S) or the Geino performer's life event is good, the Geino Producer interprets the Geino performer's imaginary narrative from the response, and shifts to (1) of influencing time.
(3) If the receiver's response to the Geino event (S) or the Geino performer's life event is bad, the Geino producer develops the different Geino event on the basis of environmental cue (happening), or develops the new Geino event (S) suitable for the receiver's imaginary narrative, and returns to (2).
(4) If the receiver's response to the Geino event (S) or the Geino performer's life event is always bad, the Geino producer makes the Geino performer retire.

5. Retirement

At 1. to 4. phase, if the situation (non-capability of image share/creation, scandal, hackneyed narrative expression technology) is continue, the Geino producer makes the Geino performer retire.

V. Conclusion

At first, this paper showed an outline model of Geino organization including the interaction between substantial and imaginary narrative, secondary explained s case analysis of Geino performer history (as substantial narrative), and finally mentioned a narrative macro strategy of artificial Geino performers with lives (as an interaction rule between substantial and imaginary narrative).

According to the Geino performer (Japanese idol) example, we can analyze that a typical form of the Geino performer macro history is "Idol image creation -> Image share by multiple media development -> Outside-environment change or scandal -> Drop or change in image." And, we can obtain a meta-history for the modeling of the Geino performer's life. And, we can model a narrative macro strategy and countermeasure cycle of the Geino producer for management of Geino performer. This strategy cycle gives a narrative macro strategy of artificial Geino performers with lives as an interaction rule between substantial and imaginary narrative

Although it is thought that Geino phenomenon is apparently performed by vulgar and non-systematical structure, by watching an interaction process between imaginary narrative creation of the receiver (individual level and mass level) and substantial narrative (Geino history, Geino work, etc.), we can obtain a systematic side (structure) clearly and a modeling of Geino information (knowledge, strategy) system.

References

[1] Ogata T, Kawamura Y (1998), The Fundamental Idea of Geino Performer's Narrative Simulation (in Japanese). Journal of Nikkei Advertising Research Institute, No.180, pp.26-31

[2] Kawamura Y, Ogata T (1999), The Modeling of the Geino Performer Life on Geino Organization Model (in Japanese). Proceedings of the 11th National Conference of Japanese Simulation & Gaming Society, pp.76-79

[3] Kawamura Y, Ogata T (1999), A Study on the Japanese Geino Organization Model and its Image Strategy (in Japanese). The Journal of Osaka University of Economics and Law, The Economic Studies Vol.23, No.2

[4] Shimaguchi M, Takeuchi H, Katahira H, Ishii J (1999), Product Development Innovation (in Japanese). Yuhikaku

[5] Wada M (1999), Interactive Marketing and Play Consumption (in Japanese) . Diamond

[6] http://www.wgn.co.jp/yuki/check.html

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Human Characteristic on Phase Discrimination between Visual and Auditory Perception by using Two-dimensional Sine Wave Stimuli

Osamu Nobuki*, Jing-Long Wu† and Yoshiro Sakai‡

*Department of Mechanical Engineering, Yamaguchi University
Tokiwadai 2-16-1, Ube, Yamaguchi 755-8611, Japan
†Department of Intelligent Mechanical Systems, Faculty of Engineering, Kagawa University
Saiwai-Cho 1-1, Takamatsu, Kagawa 760-8256, Japan
Email: wu@eng.kagawa-u.ac.jp
‡Department of Kansei Design and Engineering, Faculty of Engineering, Yamaguchi University
Tokiwadai 2-16-1, Ube, Yamaguchi 755-8611, Japan

Key Wards: *Virtual reality, Cognitive science, Human visual and auditory mechanism, Spatiotemporal characteristic*

Abstract: Human visual and auditory systems have been mainly studied as independent system, however, interactive characteristic between the visual and the auditory system was not very investigated much. On the other hand, human get information from the visual and the auditory sense, many audio-visual devices have been developed to the virtual reality and the diversification of the mass media. In order to get high reality and natural performance of these application systems, human audio-visual characteristics are need to be investigated. In this study, the phase discrimination thresholds between the visual and the auditory stimuli are measured with changing the spatiotemporal frequency of the sine-wave stimulus. The experimental results are possible to apply effective productions in the virtual reality, the drama and the mass media. Furthermore, the experimental results provided significant basic data to elucidate human audio-visual mechanism.

1. Introduction

Human visual and auditory characteristics have been separately investigated in previous studies [1][2]. There are studies discussed the mechanism of brain with using the separate characteristics of the visual and the auditory system [3][4]. The characteristics in spatial domain are to see an example of the sound source localization [5]. In application fields of the mass media and the virtual reality, the effect of the matching of sound and video has been investigated [6]. There are studies discussed about the audio-visual mutuality's related characteristic in recognition science [7]. Nevertheless, human audio-visual mutuality's related characteristic in spatiotemporal domain is needed in order to get high performance in audio-visual presenting mass media, there are few investigations argued in that domain.

In our previous studies [8][9], phase discrimination thresholds between the visual and the auditory stimulus are measured with changing of the temporal or spatiotemporal frequency of the stimuli. The results suggested that the discrimination threshold of the phase difference is getting lager and lager as increasing of the temporal frequency. And it is no longer able to distinguish of the phase difference as the temporal frequency is higher than 3 Hz. And they are able to indicate that the perception of delay between the visual and the auditory information represented as lip-sync up to this time is differentiated by the temporal frequency of audio-visual information.

In this study, luminance of visual stimulus is varied on a sin curve on the time course (sin-wave stimulus), and spatial frequency of visual stimulus (two-dimensional sin-wave grating) tapered by two-dimensional Gaussian as two-dimensional Gabor is varied as a parameter. The Gabor function is effectively used to discuss human visual characteristic [10][11]. The auditory stimulus is similar to our previous study except the pitch of the stimulus. The phase discrimination thresholds between the visual and the auditory stimulus are measured with changing the spatiotemporal frequency of the visual stimulus, and changing the temporal frequency of the auditory stimulus. The experimental results suggested that (a) the threshold of discrimination is not a monotone function when the frequency of the visual and the auditory stimulus is varied form low to high, (b) we can get a criterion about phase discrimination threshold of audio-visual information.

2. Psychological Experimental Method

2.1 Experimental System

The experimental system was proposed as similar to our previous studies [8][9]. The visual stimuli were generated by a personal computer (Gateway 2000), and displayed on a 17-in CRT (Sony CPD-17SF7). The refresh rate of the CRT was 60 Hz, and the spatial resolution was 35 pixels/degree. The CRT was calibrated with a luminance colorimeter (Minolta CS-100), and its output was linearized under software control. The auditory stimuli were generated by a soundboard (Creative Sound Blaster, PCI-128), and presented by a headphone (Sony MDR-CD370). The color condition of the visual stimuli in all the experiments was set to that the space-averaged chromaticity (CIE 1931) coordinates of the CRT were $x = 0.32$ and $y = 0.36$. The sound pressure was calibrated with a noise level meter (Rion NL-14), and its output was linearized under software control. The subjects and the personal computer had conversation with each other by using a response key. The distance between the CRT and the subjects was kept at 1.0 m by using a chin rest. All the experiments were carried out in a silent dark room.

2.2 Experimental Stimuli

As an example, one of visual experimental stimuli is shown in Fig. 1. The flame size of the square background was 6.3°×6.3°, and its luminance was set at 26.7 cd/m^2. The location of visual target stimulus was center of the square background. The visual stimulus element was a two-dimensional grating and tapered in spatial dimension, x and y directions, by two-dimensional Gaussian and its luminance function [$l(x, y, t)$] and contrast function [$c(x, y, t)$] as two-dimensional Gabor were specified as follows:

$$l(x, y, t) = l_0[1 + c(x, y, t)] \quad (1)$$
$$c(x, y, t) = c_0 \exp[-(x^2 + y^2)/s^2] \sin(2\pi fx) \sin(2\pi fy) \sin(gt) \quad (2)$$

where l_0 is mean luminance of the display (26.7 cd/㎡), c_0 is maximum contrast, f is spatial frequency, g and s are temporal frequency and the spread of Gaussian window, respectively. The Gaussian spread was fixed at 5 degrees when stimulus luminance was equal to background's one 26.7 cd/㎡. The luminance of the visual stimulus was varied from 0.970 cd/㎡ to 52.4 cd/㎡. The luminance $l[x, y, t]$ was varied in a sine curve in time course as $\sin(gt)$. The temporal frequency was set at 0.28, 0.50, 0.89, 1.6 or 2.8 Hz. The spatial frequency was set at 0.095, 0.20, 0.42, 0.89 or 1.9 c/degree.

The pitch of the auditory stimulus was set at 4000 Hz, and its sound pressure $v(t)$ was specified as follows:

$$v(t) = v_0[1 + a(t)] \quad (3)$$
$$a(t) = a_0 \sin(gt) \quad (4)$$

where v is mean sound pressure level of the stimulus 49 dB, a_0 and g are maximum amplitude and temporal frequency, respectively. The sound pressure level of the auditory stimulus was varied from 40 dB to 57 dB with a sine curve on the time course as $\sin(gt)$. The temporal frequency was set visual one too. The auditory stimulus recorded as a wave file (PCM, sampling rate 1.1 kHz, 8 bit, monaural).

The audio-visual stimuli were presented concurrently, the measurements were carried out in the cases of that the phase differentiates between the visual and the auditory stimulus were set at 0°, 30°, 60° and 90°. The phase difference means that the phase value of the visual stimulus proceeds auditory one.

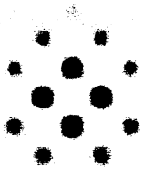


Fig. 1 An example of the two-dimensional Gabor grating as visual stimuli (two-dimensional sine-wave grating tapered in x and y direction by Gaussians, spatial frequency is 0.42 c/degree.)

2.3 Experimental Procedure

Before every experiment, the subjects had gone through a five minutes for dark adaptation and a five minutes for light adaptation. The subjects answered by using the method of three judgements (① yes, ② no and ③ can not judge) when the audio-visual stimuli were concurrently presented. The subjects were required to press the response key to indicate whether the phase between the peak of brightness at the two-dimensional Gabor grating (visual target stimulus) and the peak of sound pressure (auditory stimulus) has been recognized synchronized or not, or can not judge. That is to say, if the subjects sensed that the visual stimulus was in (out) phase with the auditory stimulus, he/she pressed the response key to answer "Yes"("No"). If the subjects could not judge that the visual and the auditory had synchronized or not, he/she pressed the "Impossible" key to indicated he/she could not judge the stimuli. The time chart of the measurements is shown in Fig. 2. The combinations of changing the spatial frequency, temporal frequency and the phase of the visual and the auditory stimulus, were 144 types. Each combinatorial stimulus was presented 16 times in a random order. Three students were served as the subjects for all the experiments.

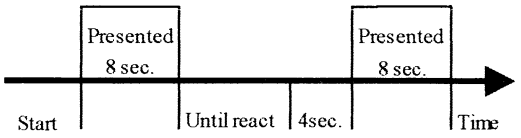


Fig. 2 Time chart of the measurements

3. Psychological Experimental Results

The average experimental results of three subjects are shown in Figs. 3 and 4. The horizontal axis indicates the phase difference between the visual and the auditory stimulus. The vertical axis indicates a ratio of the subjects reply "Yes" (the visual stimulus is in phase with the auditory stimulus) times to the total times of that the target stimuli are presented. Here, we defined this ratio as the harmony ratio. In Figs.3 and 4, SF and TF mean the spatial frequency [c/degree] and the temporal frequency [Hz] of the visual stimulus, respectively. The data in the case of that the subject pressed the "Impossible" key were excepted from the results of Figs. 3 and 4. The ratio of "Impossible" key press times to total 48 times for all subjects is tabulated in Table 1. As shown in Figs. 3 and 4, it can be detected that the decreasing of the harmony ratio depends on the temporal and the spatial frequency. These peculiarity are that the harmony ratio decreases quickly in the cases of the low spatial frequency (Fig. 3(a)) and low temporal (Fig. 4(a)), when the phase difference is increased. Conversely, the harmony ratio decreased slowly in the cases of that the spatial frequency (Fig. 3(e)) and the temporal frequency (Fig. 4(e)) are set at high values.

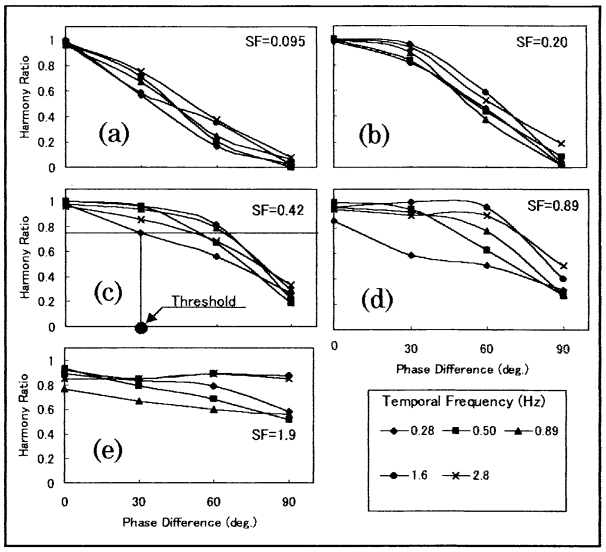


Fig. 3 Average harmony ratio of three subjects as a function of the phase difference for five cases of Spatial Frequency (SF)

From Table 1, we can say that the experimental results in the cases of temporal frequency was set at 2.8 Hz, or spatial frequency was set at 1.9 c/degree are not reliable because the ratio of the "Impossible" times to the total times is high. In the preparatory experiment, the harmony ratio could not measured if the temporal frequency is lager than 3.0 Hz. The above results suggested that the subjects are difficult to discriminate the phase difference between

the visual and the auditory stimulus if the temporal frequency is lager than 2.8 Hz or the spatial frequency is lager than 1.9 c/degree. As shown in Fig. 3(e), it can be seen that the harmony ratio is not decreased when the phase difference is increased for every temporal frequency. The tendencies of the above are similar our previous studies [9][10].

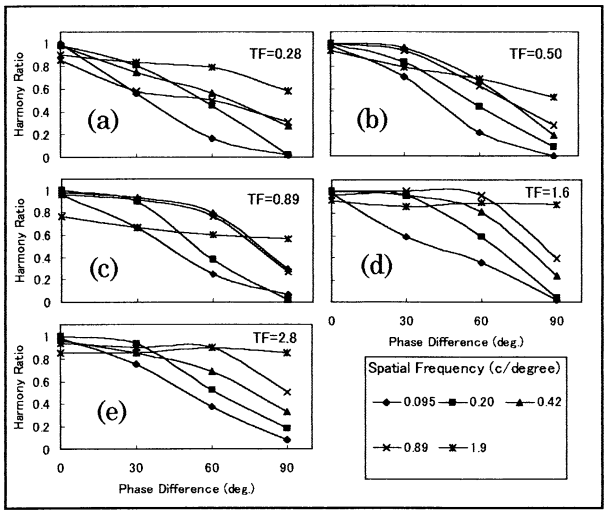


Fig. 4 Average harmony ratio of three subjects as a function of the phase difference for five cases of Temporal Frequency (TF)

Table 1 Ratio of the "Impossible" times to total 48 times measurements cross all the subjects

SF (c/deg)	TF (Hz)				
	0.28	0.50	0.89	1.6	2.8
0.095	0	0	0	0	0.083
0.20	0.021	0	0	0	0.083
0.42	0	0	0	0	0.10
0.89	0	0.021	0	0	0.19
1.9	0.10	0.083	0.13	0.19	0.25

SF: spatial frequency; TF: temporal frequency.

4. Discussion

Figures 5 and 6 show the harmony ratio as a function of the temporal frequency and the spatial frequency, respectively. They are replotted from Figs. 3 and 4, respectively. In Figs. 5(a), (b), (c) and (d), the characteristics of harmony ratio can be divided into three types as the temporal frequency is increased. The first type is that harmony ratio is nearly flat at 1, the second type shows a band-pass curve, and third type is that the harmony ratio is flat at a small value. In Fig. 6, the characteristic of harmony ratio can be divided into two types as the spatial frequency is increased. One type is that the harmony ratio is nearly flat at 1, another one is that the harmony ratio increases as the spatial frequency is increasing.

In order to investigate the detail of the characteristic of the harmony ratio, discrimination threshold of the phase difference between the visual and the auditory stimulus is defined and its calculated method can be explained as shown in Fig. 3(c). In this discussion of the discrimination threshold, the cases of that the spatial frequency was set at 1.9 c/degree and temporal frequency was set at 2.8 Hz, are excepted because their data are not reliable as shown in Table 1. The discrimination threshold of the phase difference is defined with the phase difference at the harmony ratio value is 0.75. The threshold is found at the cross point of the harmony ratio value and the 0.75 line, as shown in Fig. 3(c). In this way, the discrimination thresholds of the phase and the time lag (TPT) can be calculated, and they are summarized in Table 2.

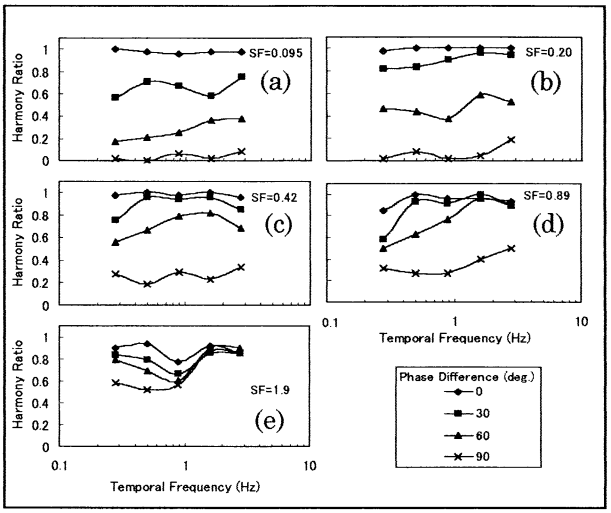


Fig. 5 Average harmony ratio of three subjects as a function of temporal frequency for four cases of phase difference

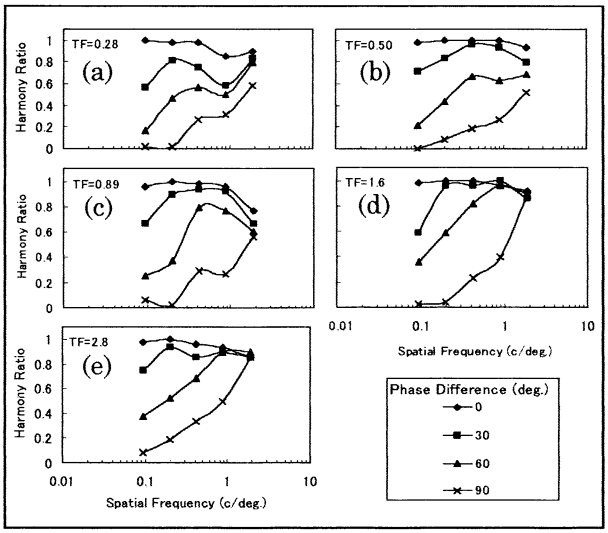


Fig. 6 Average harmony ratio of three subjects as a function of spatial frequency for four cases of phase difference

Table 2 Discrimination threshold of the phase difference and the time lag

SF (c/deg.)	TPT	TF (Hz)			
		0.28	0.50	0.89	1.6
0.095	Phase (deg.)	17	27	22	17
	Time (sec.)	0.17	0.15	0.070	0.029
0.20	Phase (deg.)	36	37	40	49
	Time (sec.)	0.36	0.21	0.12	0.085
0.42	Phase (deg.)	30	53	63	64
	Time (sec.)	0.30	0.29	0.20	0.11
0.89	Phase (deg.)	11	49	62	73
	Time (sec.)	0.10	0.27	0.19	0.13

Figure 7 shows the discrimination threshold of the phase difference as a function of the temporal (a) and the spatial (b) frequency. They

are plotted from Table 2.

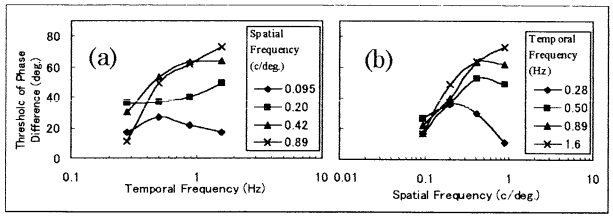


Fig. 7 Discrimination threshold of the phase difference as a function of (a) the temporal frequency and (b) the spatial frequency

Form Fig. 7(a), it can not be detected a simple tendency of discrimination threshold of phase difference as increasing of temporal frequency. In the case of that the spatial frequency is set at 0.095 c/degree (◆), the discrimination threshold of phase difference shows a band-pass characteristic of the temporal frequency. In the other cases (■, ▲ and ×), the threshold increases when the temporal frequency is increased. The results of Fig. 7(a) can be also discussed by the relationship between the threshold of the time lag and the temporal frequency as shown in Fig. 8(a).

Form Fig. 7(b), it can be detected that the discrimination thresholds (◆, ■ and ▲) of phase difference show band-pass characteristics of the spatial frequency except the case of that the temporal frequency is set at 1.6 Hz (×). That is to say, when the spatial frequency has low or high value, it is easy to discriminate the phase difference between the visual and the auditory stimulus. When the spatial frequency is set at middle value, the discrimination threshold of phase difference has large value, this means that it is difficult to discriminate the phase difference between the visual and the auditory stimulus. The reasons of above results can be considered as follows:

(1) In the range of the low spatial or the low temporal frequency, the displacement of the stimulus has large value. So, to discriminate the phase difference between the visual and the auditory stimulus is easy.

(2) In the range of the high spatial or the high temporal frequency, the density of stimulus stripes has larger value, then to discriminate the phase difference between the visual and the auditory stimulus has more chance.

(3) In the range of the middle spatial or the middle temporal frequency, the threshold of phase difference is large. The details will be investigated in our future work.

Figure 8 is plotted from Table 2, it shows the characteristics between the threshold of the time lag and the temporal frequency. The discrimination threshold of the time lag can be calculated from the threshold of the phase difference and the temporal frequency.

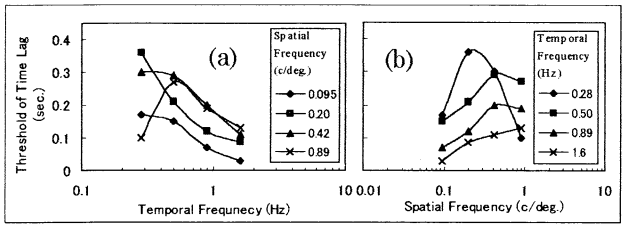


Fig. 8 Discrimination threshold of the time lag as a function of (a) the temporal frequency and (b) the spatial frequency

From Fig. 8(a), it can be detected that the discrimination threshold of the time lag decreases when the temporal frequency is increased except the case of the spatial frequency is set at 0.89 c/degree (×). From Fig. 8(b), it can be detected that when the spatial frequency is increased, the thresholds of the time lag show band-pass characteristics except the case of the temporal frequency is set at 1.6 Hz (×).

5. Summary

In this study, the thresholds of phase discrimination were measured with using two-dimensional Gabor stimuli as visual target stimulus. The experimental results suggested that the thresholds of phase discrimination are not monotone function in the spatiotemporal frequency domain. In the spatial domain, the discrimination thresholds of phase difference have a band-pass characteristic except the case of that the temporal frequency is set at large value (1.6 Hz). Conversely, in the temporal frequency domain, the discrimination threshold of the phase difference decreases when the temporal frequency is increased except the case of the spatial frequency is set at large value (0.89 c/degree). The experimental results also suggested that human can not discriminate the phase difference exactly, when the temporal frequency is larger than 2.8 Hz or the spatial frequency is larger than 1.9 c/degree.

The results of this study are possible to apply the applications of the virtual reality and the mass media. It is our future work to investigate the details of band-pass characteristic of the temporal and the spatial frequency.

References

[1] Shin'ya Nishida and Takao Sato (1992), Positive Motion After-Effect Induced by Bandpass-Filtered Random-Dot Kinematograms, Vision Res. Vol. 32, No. 9, pp. 1635-1646

[2] David A. Eddins (1999), Amplitude-modulation detection at low- and high-audio frequencies, J. Acoust. Soc. Am. 105 (2), Pt. 1, February pp. 829-837

[3] Stephen Engel, Xuemei Zhang & Brian Wandell (1997), Colour tuning in human visual cortex measured with functional magnetic resonance imaging, Nature, Vol. 388, 3 July, pp. 68-71

[4] Timothy D.Griffiths, Greaint Rees, Adrian Rees, Gary G.R. Green, Caroline Witton, Dominic Rowe, Christian Buchel, Robert Turner and Richard S.J. Frackowiak (1998), Right parietal cortex is involved in the perception of sound movement in humans, Nature neuroscience, volume 1, No. 1, May, pp. 74-79

[5] Michiko Ohkura, Yasuyuki Yanagida and Susumu Tachi (1995), Sound distance localization using virtual environment (in Japanese), SICE, Vol. 31, No. 9, pp. 1318-1323

[6] 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)

[7] Jing-Long Wu, Hiroaki Mizuhara and Yoshikazu Nishikawa (1998), Measurement of human visual and auditory characteristics for a virtual driving system, Proceedings of the International Symposium on Artificial Life and Robotics, (AROB 3th), Beppu, Oita Japan, Feb 19-21, 1998, Vol. 2, pp. 715-718

[8] Jing-Long Wu and Osamu Nobuki (1999), Human visual and auditory characteristics in the temporal frequency domain, Proceedings of the International Symposium on Artificial Life and Robotics, (AROB 4th), Beppu, Oita Japan, Feb 19-22, 1999, pp.387-390

[9] Jing-Long Wu, Osamu Nobuki and Yoshiro Sakai (1999), Human Characteristic on Phase Discrimination between Visual and Auditory Sine-Wave Stimuli, Proceedings of the 2nd International Conference on Cognitive Science (ICCS 2nd), International Conference Center, Waseda University, Tokyo, Japan, July 27-30, 1999, pp.119-124

[10] Shin'ya Nishida, Yoshio Ohtani and Yoshimichi Ejima (1992), Inhibitory Interaction in a Split/Fusion Apparent Motion: Lack of Spatial-Frequency Selectivity, Vision Res. Vol. 32, No. 8, pp. 1523-1534

[11] Tatsuto Takeuchi (1998), Visual Motion Integration in Scotopic Vision, Selection and Integration of Visual Information, Proceedings of the International Workshop on Advances in Research on Visual Cognition, Science and Technology Association and National Institute of Bioscience and Human-Technology, pp. 235-244

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Measurement of Human Visual and Auditory Calculate Characteristic by functional Magnetic Resonance Imaging (fMRI)

Hiroaki Mizuhara*, Jing-Long Wu**, Kiyoshi Negoro†, Masahiro Hashida‡,
Jun-ichi Ogasawara†, Syuichi Yamauchi‡ and Naofumi Matsunaga‡

* Department of Mechanical Engineering, Graduate School of Science and Engineering, Yamaguchi University
2-16-1 Tokiwadai, Ube City, Yamaguchi 755-8611, Japan
E-mail: c2105@stu.sv.cc.yamaguchi-u.ac.jp
** Department of Intelligent Mechanical Systems, Faculty of Engineering, Kagawa University
1-1 Saiwai-cho, Takamatsu City, Kagawa 760-8526, Japan
E-mail: wu@eng.kagawa-u.ac.jp
† Department of Neurology, Yamaguchi University School of Medicine
‡ Department of Radiology, Yamaguchi University School of Medicine
1144 Kogushi, Ube City, Yamaguchi 755-8505, Japan

Abstract

Human visual and auditory systems have high performance as complicated information processing. If the mechanisms of complicated information processing are clarified, we can construct technical systems with high performance suchlike human information processing. In previous studies, human visual and auditory systems were well investigated. However, more complicated information processing suchlike human consideration is not well clarified. In this study, the authors pay attention to mental arithmetic problems as one of the complicated information and present the mental arithmetic problems as auditory and visual inputs in functional magnetic resonance imaging (fMRI) experiments. The experimental results suggested that the human auditory calculate processing are similar to the visual calculate processing.

1. Introduction

Human visual and auditory systems have been evolving for long time and have attained very high performance. Recently, many investigations in human visual or auditory system have been reported by psychological and functional magnetic resonance imaging (fMRI) studies. In the psychological studies, mechanisms of stereoscopic and spatial attention have been frequently reported as the visual system[1-4]. And mechanisms of speech perception have also been reported as the auditory system[5-7]. In the fMRI studies, areas of color perception and that of motion perception in human visual cortex have been reported[8-10]. And cortical areas associated with native and second language are also reported[11]. However, mechanisms and cortical areas associated with more complicated information processing have been not well clarified. Furthermore, difference between visual system and auditory system of complicated information processing has been not evolved by fMRI study, although that of language processing and spatial localization has been reported by positron emission tomography (PET) studies[12, 13].

In order to clarify the mechanism of human complicated information processing, the mental arithmetic problems are used as the visual and the auditory stimuli in our study. The acquisitions of calculate ability is considered to require the immense effort and time compared with the acquirement of language[14]. Therefore, the mental arithmetic problems can be considered to suit to investigation of human complicated information processing. In our previous studies, we have measured the visual and the auditory calculate characteristics of reaction time by the psychological experiments[15-17].

In this study, we measured cortical areas of the mental arithmetic problems presented as the auditory and the visual stimuli by fMRI. The experimental results suggested that the human auditory calculate processing is similar to the visual calculate one, although that the organ of the auditory system is different from the visual system.

2. Results of Psychological Experiments

In auditory psychological experiment, experimental task was to answer the mental arithmetic problems of addition. For example, when a voice "*Twenty five plus eighty six is equals one handled twelve*" (in Japanese) was presented to subject, the subject was required to judge whether the voice problem is right or not with using two buttons to indicate his/her answer "*Yes*" or "*No*", respectively. Measurement was time from the auditory voice ended to the button pressed, and this time was defined as auditory reaction time.

The experimental results of the auditory reaction time for subject HM are shown in Fig. 1a. The horizontal axis shows value of the Difficulty of Auditory Stimuli (*DAS*). We construct a function of *DAS* in our previous studies with using the results of the auditory experiment [15-17].

In visual psychological experiment, experimental task was to answer the mental arithmetic problems of addition similarly to the auditory experiment [15-17]. In this experiment, stimuli were presented as the visual image, e.g. "25 + 86 = 111". We measured time between the image started and button pressed to indicate the subject's answer, and this measurement time was defined as the visual reaction time.

The experimental results of the visual experiment for subject HM are shown in Fig. 1b. In this figure, the vertical axis shows the visual reaction time, that is the time of judging whether "$A + B = C$" or not. The horizontal axis shows the value of difficulty of the

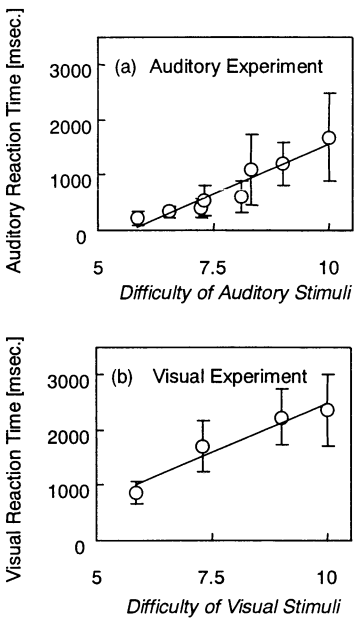


Fig. 1 Results of psychological experiments for subject HM. **(a)** Results of auditory experiment. **(b)** Results of visual experiment.

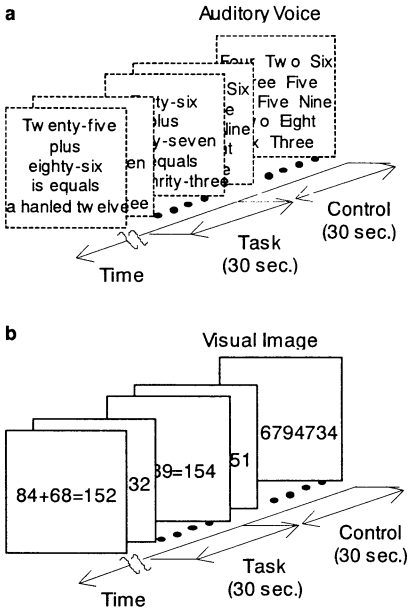


Fig. 2 Schematic illustrating of stimuli in fMRI experiments. **(a)** Stimuli were presented to subject as auditory voice. **(b)** Stimuli were presented as visual image.

visual stimuli constructed in the auditory experiment. From this figure, we can know that the auditory and the visual reaction time is increased straightly with increasing the difficulty of the auditory and the visual stimuli, respectively. Thus the difficulty of the visual stimuli can be defined as the difficulty of the auditory stimuli.

3. Experiments of fMRI

3.1 Stimuli

In the fMRI experiments, the mental arithmetic problems similar to the psychological experiments were used as auditory and visual stimuli, and we measured cortical areas of human calculate processing. The stimuli used in fMRI experiments were the mental arithmetic problems of addition as the "double figures plus double figures is equals triple figures" (for example, 56 + 75 = 131). These stimuli were the most difficult task using in the previous psychological experiments. The stimuli were presented to subjects according to the time chart as shown in Fig. 2. During first 30 [sec.], random figures were presented to subjects as auditory voice (Fig. 2a) or visual image (Fig. 2b). We defined this presentation as "control" epoch. And after them, the mental arithmetic problems were presented during 30 [sec.]. We define this presentation as "task" epoch. These task and control epochs were presented three times in one measurement. During the metal arithmetic problems were presented, subjects were required to calculate the problems in his/her head without the response buttons. Conversely, during the random figures were presented, there is no need for subjects to calculate.

Before the fMRI experiments, the rate of correct answers were measured by psychological experiments for all the subjects. In the fMRI experiments, the presenting speeds of the visual stimuli and the auditory stimuli were decided to identify the rate of correct answer for each subject with using the results of the rate of correct answers measured by the psychological experiments. The results of

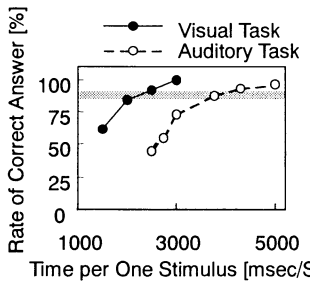


Fig. 3 Rate of correct answer for subject HM. Each datum point expressed with the opened circles (○) show the results of auditory experiment and the closed circles (●) show the results of visual experiment.

correct answer by psychological experiments for subject HM are shown in Fig. 3. The vertical axis shows the rate of correct answer [%], and the horizontal axis shows the presenting speed of stimulus [msec./stimulus]. In the fMRI experiments, the presenting speed was decided as that the rate of correct answer both of the auditory and the visual stimuli were set at 85 ~ 90 [%]. Numbers of stimuli presented in the task and the control epochs were decided by these presenting speed of the auditory and the visual stimuli.

3.2 Experimental System

In the fMRI experiments, the visual and the auditory stimuli were controlled by a personal computer (Hitachi). The visual stimuli were presented through a video monitor suchlike head mounted display (Resonance Technology), performed isolating from magnetic field. The auditory stimuli were presented through a headphone, conducted the sound with tremor of air. In all experiments, ten measurement planes were acquired with a 1.5 T MRI experimental apparatus (Siemens Inc.) with standard head coil. Echo-planar imaging was carried out with following conditions: echo time (TE) = 66 [msec.], repetition time (TR) = 1000 [msec.], slice thickness = 5 [mm], in-plain resolution = 2 [mm] × 2 [mm]. All fMRI data were analyzed by using NUMARIS software

(Siemens Inc.). In NUMARIS software, comparisons between the task and the control epochs are computed as Z statistical analyzing with using T2*-weighted MR signal for each subject.

For all the data of fMRI experiments, except the visual experiment for one subject, comparisons between the task and the control epochs were computed with threshold Z value of 1.00. In the exception, the comparison was computed with a Z value of 1.20. When the threshold Z value is set at 1.00 or 1.20, the p value is set at 0.317 or 0.231, respectively. All of these fMRI experiments had eight subjects, aged 22 ~ 25 years. And all of these experiments were performed in compliance with the relevant laws and institutional guidelines. For each subject, both of the auditory and the visual fMRI data were acquired in one experimental challenge, in order to acquire the same slice plane between the auditory and the visual experiments.

3.3 Experimental Results and Analysis

In the fMRI experiments, ten measurement planes were acquired as shown in Fig. 4. Figures 5 shows the comparison between the task epoch and the control epoch of the auditory and the visual experiments, based on group analysis using statistical analyzing. In Fig. 5, white areas indicate the significant difference of the task and the control epoch as only of the auditory, only of the visual and both of the auditory and the visual experiment for subject HM, respectively ($Z > 1.0, p < 0.317$). Significant activation in posterior inferior parietal lobe (Brodmann Area 39: BA39) appeared in the auditory and the visual experiments for subject HM (Figs. 5 (1)-①). These activation in left posterior inferior parietal lobe were observed in all eight subjects for both of the auditory and the visual experiments, and these activation in right posterior inferior parietal lobe were observed in six subjects for the auditory experiment and in seven subjects for the visual experiment, respectively. Furthermore, the activation in left posterior parietal lobe (BA40) was also observed in all the eight subjects for the auditory experiment and in seven subjects for the visual experiment. The activation in right posterior parietal lobe was observed in two subjects for the auditory and the visual experiments, respectively (Figs. 5 (1)-②). When these cortical areas (BA39 and BA40) were injured, patients were reported as trouble to calculate[14, 18]. Namely, these cortical areas are mainly considered as the areas of human calculation. Furthermore, these cortical areas are also well known as the comprehension of language[11, 13 and 14].

We also observed activation in mid limbic lobe (BA24) for the auditory experiment (all the eight subjects) and for the visual experiment (six subjects) (Figs. 5 (2)-③). As this cortical area, investigations were reported as concern with attention[19, 20]. Activation in left superior frontal lobe (BA9) was observed in all the eight subjects for the auditory experiment and in seven subjects for the visual experiment (Figs. 5 (2)-④). And activation in right superior frontal lobe was also observed in four subjects for the auditory experiment and in seven subjects for the visual experiment. These cortical areas were considered to realize the parts of short term memory[18, 21]. Furthermore, activation in left and right posterior inferior frontal lobe (BA44) were observed in seven subjects for the auditory experiment and in all the eight subjects for the visual

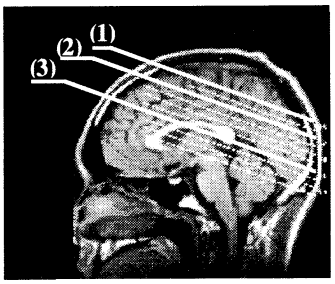


Fig. 4 Slice position of brain for subject HM in the fMRI experiments

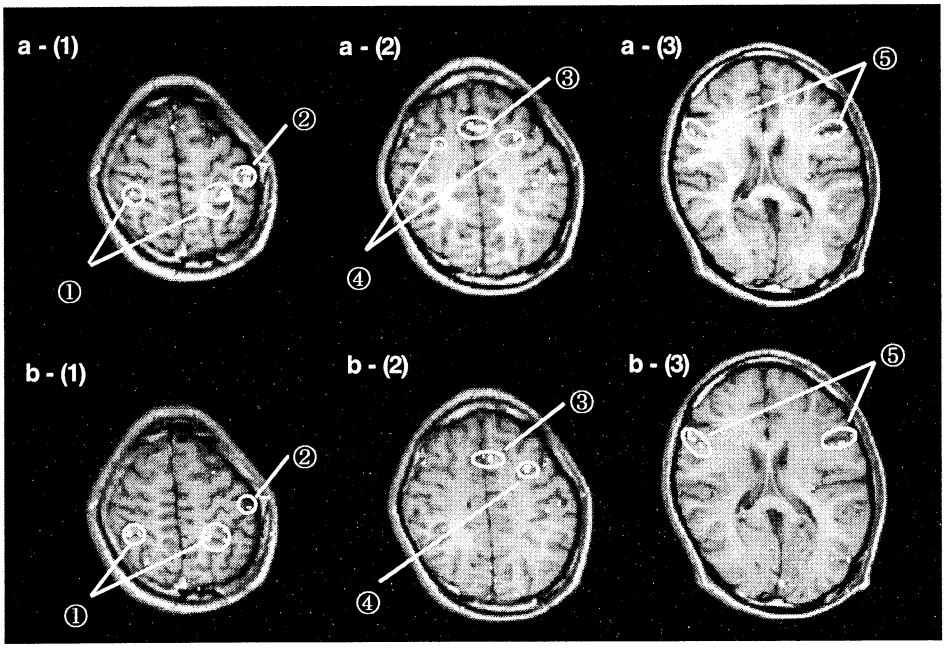


Fig. 5 Results of activation as the mental arithmetic problems for subject HM. **(a)** Results of the auditory experiment. **(b)** Results of the visual experiments. **(1)** Top of ten measurement planes as shown in **Fig. 4**. **(2)** Third plane from top of ten measurement planes. **(3)** Seventh plane from top of ten measurement planes.

experiment (Figs. 5 (3)-⑤). These cortical areas are well known as the description of language[11, 14 and 18].

In previous investigations, the followings are considered as causes of trouble to calculate[14]. The trouble of calculation is caused by the trouble as comprehension and description of language. When rule of calculation (e.g. advance of figures) are troubled, the calculation are troubled. When the advanced figure are obliterated, the calculation is also troubled. And when attention are obstructed, the trouble of calculation are caused. In the results of fMRI experiments, we observed activation of the parts of comprehension of language, description of language, calculation, and attention and short term memory. Namely, these results of fMRI experiments well agree to the results of the cases as trouble of calculation.

4. General Discussion

The results of the psychological experiments in our previous study suggested that the mechanism of the auditory calculate processing is similar to the visual calculate one, although that the organ of the auditory system is different from the visual one.

In order to evidence this consideration (the mechanism of the auditory calculate processing is similar to the one of the visual calculate processing), we discuss the difference between the auditory and the visual calculate processing with using the results of fMRI experiments. The results for both of the auditory and the visual experiments showed similarly activation of the parts of comprehension of language, description of language, calculation, attention and short term memory. Namely, these results suggested that the auditory calculation is similarly processed as the visual calculation. However, volume of the activation were not equal to each other, although the difficulty of calculation tasks are equaled with using the results as the rate of correct answer. Thus the capacity of the auditory calculate processing is not equal to one of the visual calculate processing, although the auditory calculation is similarly processed as the visual calculation. These results of fMRI experiments well agree to one of the psychological experiments in our previous studies.

5. Conclusion

The results of our previous psychological experiments and the fMRI experiments in this study, suggested that the mechanism of the auditory calculate processing is similar to one of the visual calculate processing, although that the organ of the auditory system is different from the visual one. However, the capacity of the auditory calculate processing and the mechanism of the auditory information processing except the calculation may be not equals to the capacity of the visual calculate processing and the mechanism of the visual information processing. The details will be investigated in our future works. Human calculation can be considered as one of the processing of human consideration. Namely, this investigation will be significant to investigation of human consideration. If the mechanism of human consideration is clarified, technical systems with high performance suchlike human consideration will be constructed, with applying the mechanism of processing of human consideration to the algorithm of technical systems.

Reference

1. J. Malik, B. L. Anderson and C. E. Charowhas, "Stereoscopic occlusion junctions", *Nature Neurosci.*, **Vol.2**, pp.840 – 843 (1999).
2. J. M. Harris, S. P. Mckee and S. N. J. Watamaniuk, "Visual search for motion-in-depth: stereomotion does not 'pop out' from disparity noise", *Nature Neurosci.*, **Vol.1**, pp.165 – 168 (1998).
3. J. S. Joseph, M. M. Chun and K. Nakayama, "Attentional requirements in a 'preattentive' feature search task", *Nature*, **Vol.387**, pp.805 – 807 (1997).
4. H. Mizuhara, J. Wu and Y. Nishikawa, "The degree of human visual attention in the visual search", *Artificial Life and Robotics (Springer)*, **Vol.4**, in press (2000).
5. C. S. Throckmorton and L. M. Collins, "Investigation of the effects of temporal and spatial interactions on speech-recognition skills in cochlear-implant subjects", *J. Acoust. Soc. Am.*, **Vol.105**, pp.861 – 873 (1999).
6. A. Cheveigne, "Missing-data model of vowel identification", *J. Acoust. Soc. Am.*, **Vol.105**, pp.3497 – 3508 (1999).
7. A. Sharma and M. F. Dorman, "Cortical auditory evoked potential correlates of categorical perception of voice-onset time", *J. Acoust. Soc. Am.*, **Vol.106**, pp.1078 – 1083 (1999).
8. S. Engel, X. Zhang and B. Wandell, "Color tuning in human visual cortex measured with functional magnetic resonance imaging", *Nature*, **Vol.388**, pp.68 – 71 (1997).
9. S. Engel et al., "fMRI of human visual cortex", *Nature*, **Vol.369**, pp.525 (1994).
10. J. B. Reppas et al., "Representation of motion boundaries in retinotopic human visual cortical areas", *Nature*, **Vol.388**, pp.175 – 179 (1997).
11. K. Kim, N. Relkin, K. Lee and J. Hirch, "Distinct cortical areas associated with native and second languages", *Nature*, **Vol.388**, pp.171 – 174 (1997).
12. K. O. Bushara et al. "Modality-specific frontal and parietal areas for auditory and visual spatial localization in humans", *Nature Neurosci.*, **Vol.2**, pp.759 – 766 (1999).
13. S. E. Petersen et al. "Positron emission tomographic studies of the cortical anatomy of single-word processing", *Nature*, **Vol.331**, pp.585 – 589 (1988).
14. A. Yamadori, "Introduction to neuropsychology (in Japanese)", *IGAKU-SHOIN* (1985).
15. J. L. Wu, H. Mizuhara and Y. Nishikawa, "Measurement of human visual and auditory characteristics for a virtual driving system", *Proc. of The Third Int. Symp. on Artificial Life and Robotics (AROB '98)*, pp.715 - 718 (1998).
16. J. L. Wu, H. Mizuhara and Y. Nishikawa, "On human related characteristic between visual search and speech perception (in Japanese)", *Transaction of the Institute of Systems, Control and Information Engineers*, **Vol.12**, pp.98 - 105 (1999).
17. J.L. Wu, H. Mizuhara and Y. Nishikawa, "Human characteristics of parallel information processing on human visual and auditory systems", *IEEE Systems, Man and Cybernetics (SMC '99)*, pp. II 54 - II 59 (1999).
18. M. Iriki and K. Toyama, "Physiology (in Japanese)", *BUNKODO* (1986).
19. S. Jackson, R. Marrocco and M. I. Posner, "Networks of anatomical areas controlling visuospatial attention", *Neural Networks*, **Vol.7**, pp.925 – 944 (1994).
20. M. I. Posner, S. E. Petersen, P. I. Fox and M. E. Raichele, "Localization of cognitive operations in the human brain", *Science*, **Vol.240**, pp.1627 – 1631 (1988).
21. E. Koechlin et al., "The role of the attention prefrontal cortex in human cognition", *Nature*, **399**, 148 – 151 (1999).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Subjective Contours Strength depends on the Product of the Length of the Subjective Edges

Sheng Ge, Jing-Long Wu† and Shunichi Kawano

Department of Mechanical Engineering, Yamaguchi University
Tokiwadai 2-16-1, Ube, Yamaguchi 755-8611, Japan
†Department of Intelligent Mechanical Systems, Faculty of Engineering, Kagawa University
Saiwai-Cho 1-1, Takamatsu 760-8526, Japan
E-mail: wu@eng.kagawa-u.ac.jp

Abstract: Three experiments were designed in order to find the mathematical characteristic of the strength of subjective contours. In Experiment 1, we inspected the apparent strength of subjective contours under the condition that the experimental figures were dissymmetric. It was founded the mean apparent strength of subjective contours was a direct proportion to product of the length of the subjective edges. In Experiment 2 another kind of experimental figures were used and the experimental results confirmed that the conclusion of Experiment 1 was tenable. In Experiment 3, the figures including side-elements were used. It confirmed that the conclusions of Experiments 1 and 2 were still suit for such case.

Keywords: *Cognitive science, Human visual subjective contours, Subjective contours strength, Inducing elements, Pattern recognition.*

1. INTRODUCTION

Shipley & Kellman [1] found that the interpolation which determined by both the size of inducing elements and the spacing between inducing elements effected the subjective contours strength. They used three-fourths-circle as the inducing elements and varied the size and the spacing between inducing elements independently. They introduced a variable V which stood for the fraction of the whole edge that is physically. Then the mean subjective contours strength rated as a function of the variable V. However, all of the displays they used were symmetric figures. Do the results they found still suit for dissymmetric figures? These questions became to be the starting point of our study.

2. PSYCHOLOGICAL EXPERIMENTS

2.1 Experiment 1

2.1.1 Subjects Ten subjects (9 male, 1 female) who were experienced psychophysical observers but ignored the purpose of the experiment were randomized enrolled into our study group. All of them had previously practice before the experiment in order to make them understanding subjective contours and being familiar with judging the subjective contours strength. And they all had a normal or corrected-to-normal visual acuity. Subjects ranged in age from 21 to 36 years.

2.1.2 Apparatus All the experiments were done in a darkroom. A personal computer (NEC PC-9812) was used to produce the stimuli displays and a SONY RGB monitor (CPD-17SF7) which measured 20×25 cm and set as middle resolution mode (800×600 pixels) was used to display the stimuli. All the stimuli were presented to each subject in a random order. A chin rest was used to keep the subjects head stationary and aligned with the screen. Subjects viewed the stimuli binocularly from a distance of 92 cm. The stimuli were a series of standard Kanizsa triangle [2], the inducing elements were white (40.5 cd/m^2) and the background was gray (16.5 cd/m^2). The basic standard figure was 5.6 mm (20 pixels, 0.35°) of inducing elements size (radius) and 14.6 mm (52 pixels, 0.91°) of spacing. The ratio defined as the interpolated edge to the entire edge was a constant of 0.43. All the triangles in the experimental figures (Fig.1) had a deviation in direction and distance relative to the basic standard figure. As shown in Fig. 1, the direction in which the triangle deviation was set at 0° and 180°, 90° and 270°, 45° and 225°, 135° and 315°, respectively. In every respective experiment, the triangle deviated in one of the above eight directions. The distance in which the triangle deviation was set at: 0, 2, 4, 6, 8, 10, 12, 14, 16, 18 and 20 pixels, respectively.

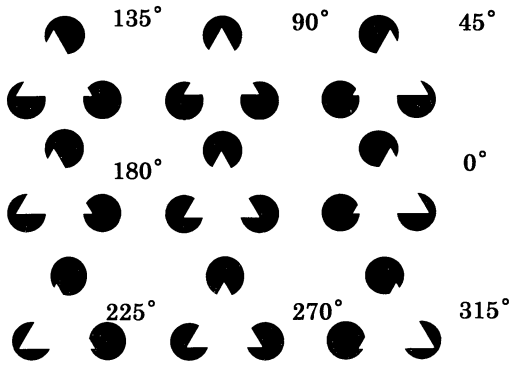


Fig.1 Typical stimuli figures of Experiment 1

2.1.3 Procedure A magnitude estimation procedure [3] was used to rate the strength of the subjective contours. And subjects were inquired to evaluate the subjective contours strength of experimental figures compared to the basic standard figures from rating 0 to rating 10 by clicking the bottoms. An example of the experimental figures is shown in Fig. 2. The displays would not be removed until the subject responded. And there was a five seconds interval before the next stimulus was displayed, during this period the screen was gray (16.5 cd/m^2) and none stimulus was shown.

2.1.4 Results and Discussion Figure 3 showed the mean

subjective contours strength plotted as a function of the deviation in four directions. Obviously, subjective contours strength is clarity when the deviation is smaller. The graph also showed that the function relationship is practically the same one although they are different in deviation direction. It showed that this function relationship is independent on deviation direction.

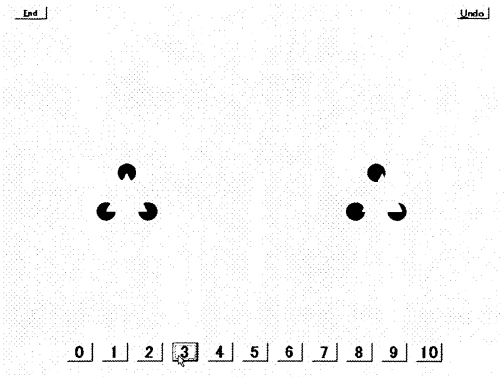


Fig. 2 A captured screen of Experiment 1. The left figure is a standard figure, the right figure is an experimental figure. The subjective contours strength of the standard figure is defined as 10.

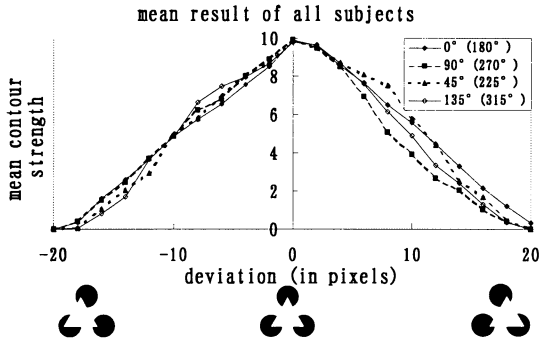


Fig. 3 Mean contour strength for all directions is plotted as the same function of deviation in Experiment 1.

Shipley & Kellman [1] defined the $V = 2R/(2R+d)$ as the fraction of the whole edge that is physically specified. In Experiment 1, all the experimental figures have a deviation in a certain direction, so the figures are dissymmetric, then the "whole edge that is physically specified" has no clear-out definition here. We proposed the definition of "the fraction of whole edge that is physically specified" is the ratio of the sum (or the average) of whole edges physically specified to the sum of the circumference of the *subjective figure* [4]. In Shipley & Kellman's experiment [1], they used symmetric figure displays, so

$$V = \frac{8R}{4(2R+d)} = \frac{2R}{2R+d} \qquad (1)$$

Using the above explain, the ratio in our Experiment 1 will be defined as

$$V_1 = \frac{L_1 + L_2 + L_3 + L_4 + L_5 + L_6}{3(2R+d)} \qquad (2)$$

The definition of L_1 to L_6 and R are shown in Fig. 4. We inspected the relationship between the contour strength and variable V_1 (Fig. 5), obviously, it is not a linear relationship.

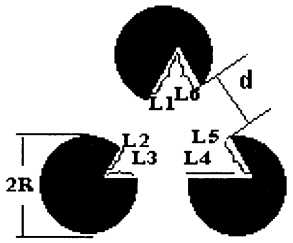


Fig. 4 Definition of the variables L_1, L_2, L_3, L_4 , L_5, L_6, R and d

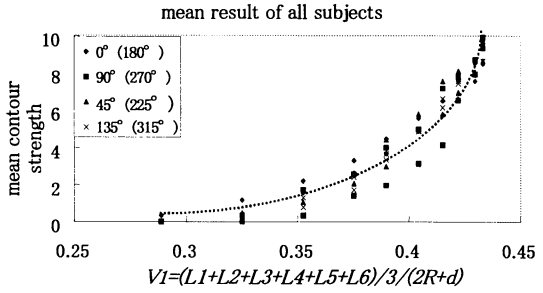


Fig. 5 Mean contour strength for all directions plotted as a nonlinear function of the variable $V_1 = (L_1 + L_2 + L_3 + L_4 + L_5 + L_6)/3/(2R + d)$.

According to the above results we thought the ratio V which defined by Shipley & Kellman [1] is not suitable for our experimental results. We noticed that when deviation overpast or just overpast the radius, at least one vertex of the triangle is out of the inducing elements, and at this time the mean subjective contours strength is zero approximately. Figure 4 is a typical experimental figure, we marked the length of edges of interior angles with L_1 to L_6. Here, we called this edge as ***subjective edge***. When the vertex of triangle is out of the inducing elements, there is at least one *subjective edge* became to be zero. It is naturally to presume that the mean subjective contours strength is a direct proportion to product of the length of the *subjective edges*. So we introduce a ratio

$$V_2 = \frac{L_1}{R} * \frac{L_2}{R} * \frac{L_3}{R} * \frac{L_4}{R} * \frac{L_5}{R} * \frac{L_6}{R} = \frac{L_1 * L_2 * L_3 * L_4 * L_5 * L_6}{R^6} \qquad (3)$$

the definition of L_1 to L_6 and R are shown in Fig. 4. Where, Li (i = 1, 2, ..., 6) is the length of subjective edges. R is the original length of the subjective edges, i.e. when the deviation is zero, the length of subjective edges are equate to R. The horizontal axis of Fig. 6 represented the ratio V_2 and the vertical axis represented the subjective contours strength. It showed that mean subjective contours strength is a function of ratio V_2, the function relationship is approximately linear. The best fitting linear regression line of Fig. 6 has an intercept of 0.79 with a slope of 8.81 (r = 0.98).

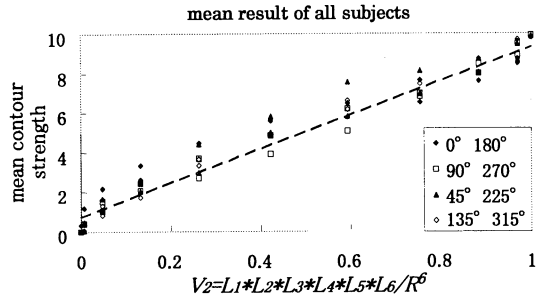


Fig. 6 Mean contour strength for all directions plotted as a function of the variable $V_2=L_1*L_2*L_3*L_4*L_5*L_6/R^6$

2.2 Experiment 2

2.2.1 Subjects Five subjects of Experiment 1 and five new subjects ranged in age from 21 to 36 years.

2.2.2 Apparatus and Procedure The apparatus used was almost the same as those used in Experiment 1. But the stimuli figures were not standard Kanizsa triangle but four three-fourths-circle (Fig.7), it was 5.6 mm (20 pixels, 0.35°) of inducing elements size (radius) and 11.3 mm (40 pixels, 0.70°) of spacing, the ratio defined as the interpolated edge to the entire edge was a constant of 0.5. The procedure used was the same to those used in Experiment 1.

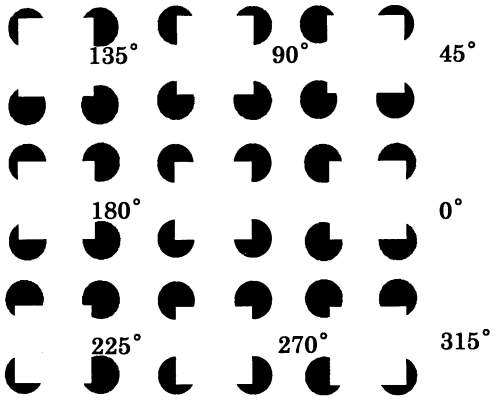


Fig.7 Typical stimuli figures of Experiment 2

2.2.3 Results and Discussion Figure 8 showed that the function relationship between subjective contours strength and deviation is the same as that in Experiment 1 although the experimental figures of Experiment 2 are different to those used in Experiment 1.

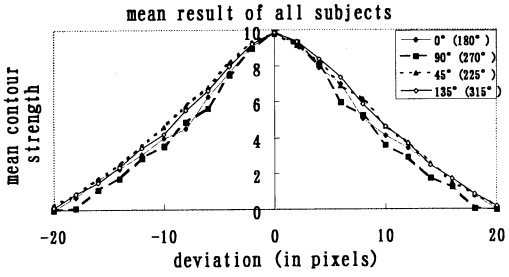


Fig. 8 Mean contour strength decreased when deviation is increased. And for all directions it plotted as the same function of deviation in Experiment 2.

As Experiment 1 we introduce a variable

$$V_3 = \frac{L_1 * L_2 * L_3 * L_4 * L_5 * L_6 * L_7 * L_8}{R^8}. \quad (4)$$

V_3 is defined as the product of the ratios of every subjective edge to inducing element size (radius). The definition of L_1 to L_8 and R is shown in Fig. 9.

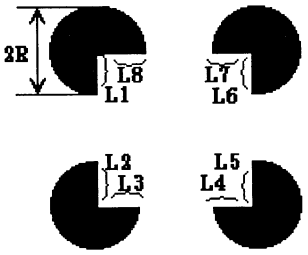


Fig. 9 Definition of the variable L_1, L_2, L_3, L_4 , L_5, L_6, L_7, L_8 and R

Figure 10 confirmed that the definition of ratio in Experiment 1 is tenable in Experiment 2. The best fitting linear regression line of Fig. 10 has an intercept of 1.14 with a slope of 8.40 (r = 0.98).

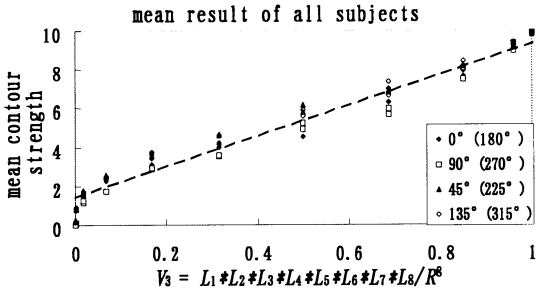


Fig. 10 Mean contour strength for all directions plotted as a function of the variable $V3 =(L_1 * L_2 * L_3 * L_4 * L_5 * L_6 * L_7 * L_8) / R^8$

2.3 Experiment 3

2.3.1 Subjects and Apparatus The subjects were same as in Experiment 2. The apparatus used was almost the same as those used in Experiment 1.

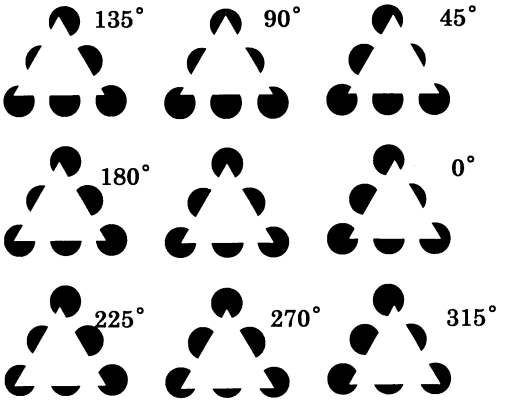


Fig. 11 Typical stimuli figures of Experiment 3

Differed from Experiment 1, side-elements were inlaid (Fig. 11). It was 5.6 mm (20 pixels, 0.35°) of inducing elements size (radius) and 4.98 mm (18 pixels, 0.31°) of spacing, the ratio defined as the interpolated edge to the entire edge was a constant of 0.70.

2.3.2 Results and Discussion Figure 12 showed that the function relationship between subjective contours strength and deviation is the same as that in Experiments 1 and 2 although the experimental figure of Experiment 3 contained side-elements.

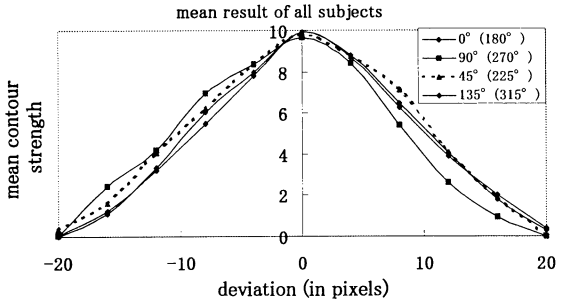


Fig. 12 Mean contour strength decreased when deviation is increased. And for all directions it plotted as the same function of deviation in Experiment 3.

Same as Experiments 1 and 2, we introduce a variable

$$V_4 = \frac{L_1 * L_2 * L_3 * L_4 * L_5 * L_6}{R^6} * \frac{l_1 * l_2 * l_3}{(2R)^3} . \quad (5)$$

V_4 is defined as the product of the ratios of every subjective edge to inducing element size (radius). The definition of L_1 to L_6, l_1 *to* l_3 and R are shown in Fig. 13. Using V_4, Fig. 12 can be changed to Fig. 14.

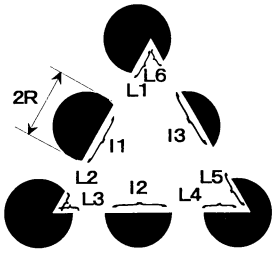


Fig.13 Definition of the variables L_1, L_2, L_3, L_4, L_5, L_6, l_1, l_2, l_3 and R

The best fitting linear regression line of Fig 14 has an intercept of 1.19 with a slope of 8.81 (r = 0.97).

The experimental results confirmed that the conclusions of Experiments 1 and 2 are still suit for Experiment 3 although the experimental figures contained side-elements.

3. GENERAL DISCUSSION

Based on the results of Experiments 1, 2 and 3, a general conclusion could be drawn, i.e., subjective contours strength is a direct proportion to product of the length of the *subjective edges.* Here, we defined the subjective contours strength as *S.*

In Experiments 1 and 2, it can be found:

$$S \propto \prod_{i=1}^{n} \frac{Li}{R} \quad (6)$$

Where *Li* is the length of *subjective edges, R* is the size of inducing elements (radius), *n* stands for the sum of the *subjective edge.*

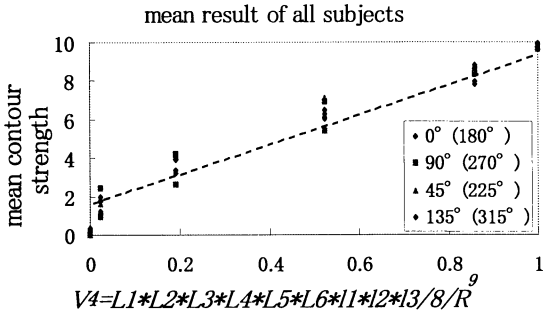


Fig. 14 Mean contour strength for all directions plotted as a function of the variable $V4 = (L_1 * L_2 * L_3 * L_4 * L_5 * L_6 * l_1 * l_2 * l_3) / (8R^9)$

However, in Experiment 3, we can find:

$$S \propto \prod_{i=1}^{6} \frac{Li}{R} * \prod_{j=1}^{3} \frac{lj}{2R} \quad (7)$$

Where *Li* and *lj* are the length of *subjective edges, R* is the size of inducing elements (Not the side-elements) and 2*R* is the size of inducing elements (side-elements).

So we introduce a new ratio *Vnew* which defined as Equation 8:

$$Vnew = \prod_{i=1}^{m} \frac{Li}{Is} \quad (8)$$

Where *Li* is the length of *subjective edges, Is* stands for the size of inducing elements (In our experiment, the "size" means radius), and *m* is the sum of the *subjective edge.* So a general conclusion can be drawn as Equation 9:

$$S \propto Vnew \quad (9)$$

That is to say, the strength of the subjective contours can be explained by the quantitative mathematical equation. Based on this finding, a new algorithm of pattern recognition will be proposed in our future work.

REFERENCES

[1] SHIPLEY,T. & KELLMAN,P.(1992). Strength of interpolated illusory contours depends on the ratio of physically specified to total edge length. *Perception & psychophysics*, **52,** 97-106.

[2] KANIZSA, G. (1955). Margini quasi-percettivi in campi con stimolazione omogenea. *Revista di Psicologia* ,**49**, 7-30.

[3] DAY, R. H. & KASPERCZYK, R. T. (1983) A modal completion as a basis for illusory contours. *Perception and Psychophysics*, **33**, 355-364.

[4] WALLACH, H., & SLAUGHTER, V. (1988). The role of memory in perceiving subjective contours. *Perception & Psychophysics*, **43**, 101-106.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Designing a Pattern Generator for a Walking Robot by Modeling the Human Creative Process

Inamoto, T., Murao, H., Kryssanov, V.V., Kurematsu, Y., and Kitamura, S.
Dept. Computer and Systems Engineering, Faculty of Engineering, Kobe University,
Rokko-dai, Nada-ku, Kobe 657-8501, Japan

Abstract

This paper describes designing a pattern generator for a walking robot by applying genetic algorithms (GA), which is made based on a semiotic theory of creativity. In terms of algebraic semiotics introduced by Joseph Goguen, the design creative process is formalized as a sequence of mappings within and between sign systems that represent the design space at different stages of the design process. The relationship between the theory and the application of GA as a practical way to design engineering systems is shown. Mappings within a sign system correspond to the evolutionary search within a design space defined by a given genotype, and mappings between sign systems – to the search for the best design space. To model the main mechanisms of creativity, a co-evolutionary algorithm is applied, which is comprised of two groups of GA. One group consists of creatures encoding a pattern generator, while another consists of creatures encoding a fitness function for the former. A case study is made, and conclusions are given.

1 Introduction

In the last few decades, many people have empirically and theoretically studied creativity in various fields of science, such as psychology and philosophy. Often, the motivation for such great interest was the wish to improve the understanding of creativity in order to develop computer tools, which would be capable of designing innovative products. There have been many works on design creativity reported in the literature, and many models of the creative process have been proposed [1].

This paper describes the application of a model of the creative process, that was originally introduced in [2], to construct an Artificial Neural Network (ANN), which is to generate a rhythm for biped trajectory. The main point of the approach is to use the framework of co-evolutionary GA in order to enable searching for a larger problem space, enhancing in this way the designer's creative abilities.

2 Designing the walking pattern generator

2.1 The model

The task of designing a walking pattern generator for a robot can be considered as a task of constructing an Artificial Neural Network, which generates a rhythm suitable to walk [3]. The ANN is developed by combining neurons in such a way that each of the neurons has a continuous inner-state, and its behavior is represented by means of two first-order differential equations with two variables. The mathematical model of the ANN is as follows (see [3] for details):

$$\begin{aligned} T_a \frac{dx_i(t)}{dt} + x_i(t) &= -\sum_{j=0, j\neq i}^{n-1} a_{ij} y_i(t) \\ &\quad - b f_i(t) + s_i \\ T_f \frac{df_i(t)}{dt} + f_i(t) &= y_i(t) \\ y_i(t) &= \begin{cases} x_i(t) & \text{if } x_i(t) \geq 0 \\ 0 & \text{otherwise} \end{cases} \\ i &= (0, 1, \cdots, n-1) \end{aligned} \tag{1}$$

We will consider that a model of the robot, which uses a pattern generated by the ANN, includes four joints and four links, and the number of neurons of the ANN is equal to 4 (according to the number of the robot joints). The correspondence between the robot and the ANN is shown in Figure 1.

2.2 The generation of walking patterns by a single GA

The ANN specified by (1) can generate various rhythms by appropriately adjusting the parameters. In [4], the author has utilized GA to determine the connection weights between the neurons of the ANN. Other parameters of the model have been calculated as described in [5], and they are shown in Table 1.

To apply GA to the considered problem, it is needed to define a fitness function that is to evaluate differ-

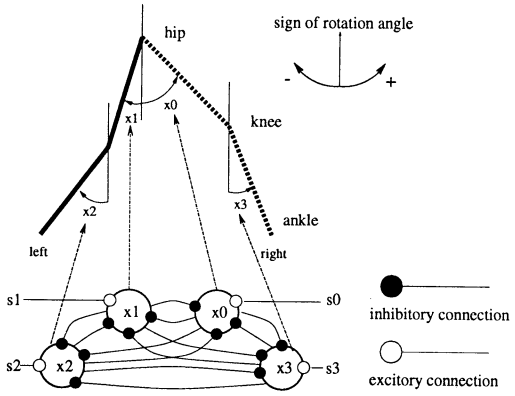


Figure 1. The robot and the ANN.

Table 1. ANN parameters.

name of the parameter	T_a	T_f	b	s_0, s_3	s_1, s_2
its value	1	12	2.5	0.3	1.0

ent walking patterns generated. As there exists no universal criteria for the latter, the function should be determined empirically as a weighted summation of a few evaluations. The weights of the fitness function are critical for the behavior of GA, and a wrong choice of the weights leads to early convergence that prevents from obtaining good results. So, to implement GA suitable for the generation of good walking patterns, it is necessary to determine the weights that is a generally difficult task for a human designer. We will assume that solving this task requires possessing some creativity.

3 Semiotic approach to modeling the designer's creative process

3.1 Human creativity

It is now widely recognized that there are two types of human creativity. The first type is called *improbabilist*: a new product is created by making appropriate connections between already existing elements. The product is considered creative because people, at the beginning, think that designing such a product seems improbable. The second type is called *impossibilist*: a new product is created by transforming the conceptual space corresponding to the design space. The product is creative because it is impossible to design the product from the original conceptual space. *Improbabilist* creativity is based on operations within an identical conceptual space, so the two types of creativity are mutually compensative rather than contrary to each other. In fact, human creativity is realized as the interaction of both types of creative thinking.

By Graham Wallas [6], the creative process is divided into four phases: *preparation*, *incubation*, *illumination*, and *verification*.

Such division emphasizes the rôles of divergent thinking and convergent thinking, which are primary cognitive mechanisms in human creative activity. Divergent thinking is the mechanism to produce creative ideas, and convergent thinking is to logically evaluate candidate ideas and select the best one among them. It is believed that the former mechanism plays the prime rôle during the *incubation* phase, and the second – during the *preparation* and *verification* phases. It is readily possible to relate Wallas' model to design synthesis and analysis in engineering. However, merely iterating some procedures of synthesis and analysis is not enough to generate a creative solution. Divergent thinking in Wallas' model includes aspects of both *improbabilist* and *impossibilist* creativity, and it is not restricted to be in one conceptual space. It was shown that in a general case, synthesis corresponds to the transformation of a conceptual space, while analysis corresponds to a search across the conceptual space [2].

3.2 Formalization of the creative process

From the above discussion, it can be seen that modeling creativity is closely related to modeling conceptual spaces and operations on them. Let us formally define a design conceptual space in terms of algebraic semiotics.

A conceptual space (i.e. design space) can be represented as a sign system Ξ:

$$\Xi =< S, V, C, R, A >,$$

where S is a set of sign sorts, V is a set of data sorts, C is a set of constructors, R is a set of relations and functions, and A is a set of axioms. There are partial orderings on sorts (sort-subsort) and constructors (by levels) in the sign system. The constructors of each levels are also partially ordered (priority ordering) [7].

The creative process can be described in terms of morphisms between sign systems. A semiotic morphism $M : \Xi_1 \mapsto \Xi_2$ represents a mapping that maps sign sorts, data sorts, constructors, relations and functions, and axioms of the sign system Ξ_1 to sign sorts, data sorts, constructors, relations and functions, and axioms of the sign system Ξ_2, while preserving the subsort ordering. The design creative process can be

represented as a recursion of morphisms defined on a sign system as follows [2]:

$$f_k(M_k(f_{k-1}(M_{k-1}(\cdots M_1(f_0(\Xi_0))\cdots)))) \mapsto \Omega,$$

where Ω denotes a design model resulted from the creative process, and Ξ_0 corresponds to a conceptual space, which includes an initial specification of the designed product. $M_i(\Xi) = \Xi'$ is a semiotic morphism that maps a sign system Ξ to a distinct sign system Ξ' and transforms the conceptual space. $f_i(\Xi)$ is a composition of semiotic morphisms that represents the integral effect of the synthesis and analysis operations:

$$f_i = \Delta^-(\Xi_i') \circ \Delta^+(\Xi_i)$$

where $\Delta^+(\Xi) = \Xi'$ is the *semiotic divergence* operator such that if $\Xi = <S, V, C, R, A>$, then $\Xi' = <S \cup S, V \cup V', C \cup C', R, A>$, $S' \neq \emptyset$ and $C' \neq \emptyset$; $\Delta^-(\Xi') = \Xi''$ is the *semiotic convergence* operator: if $\Xi' = <S, V, C, R, A>$, then $\Xi'' = <S, V, C, R \cup R', A \cup A'>$, $R \neq \emptyset$ and/or $A' \neq \emptyset$.

4 Implementation

In terms of GA, *impossibilist* creativity, which is more general and based on the transition between different conceptual spaces, corresponds to the changing of the problem space by reconsidering the genotype and the objective function. The later, however, cannot be modeled by a single GA, but a co-evolutionary GA can be employed. The co-evolutionary GA consists of two algorithms. The first generates an ordinary group of creatures, where each of them encodes a solution candidate (e.g. walking pattern). The second is an algorithm, where for each of the creatures, the phenotype serves as a fitness function for the creatures of the first algorithm. We will call the first GA as *GAmain*, and the second – as *GAfit*.

Some of the characteristics of *GAmain* were described in Section. 2.2, while other characteristics of the algorithm are given below:

- The fitness function is a weighted sum of four parameters: *walk distance*, *max step*, *pivot changing count*, and *summation of angles*. After normalization of the parameters in the range $[0, 1]$, they are combined into the fitness function with four weights.
- The creature also has a number in phenotype which represents the number of fitness functions to be used by the creature. If this number is N, it randomly selects N creatures from *GAfit* and gets N real values as the product of the evaluations of the creature and the phenotype in *GAfit*. Then, it sets the fitness as the mean of these values.

Table 2. Simulation parameters.

name of parameter	type of GA	
	GAmain	*GAfit*
generation	30	40
population	100	100
mutation ratio	0.00015	0.00015
crossover ratio	0.6	0.6

Next we describe characteristics of *GAfit*.

- The phenotype consists of four real numbers which are selected so that the sum of them to be equal to 1. This is the fitness function for the creatures in *GAmain*.
- The fitness of a creature in *GAfit* is the total count of time when it has been used by the creature which succeeded in having children at *GAmain*.

In brief, the process of the co-evolution with GA can be described as follows:

1. (Randomly) generate two initial populations.
2. Run *GAmain*. The fitnesses of creatures in *GAfit* increases.
3. Generate a new group of *GAmain* and immigrate elites at the last generation of the older group to the newer group.
4. Apply the genetic operators to the creatures of *GAfit*.
5. Repeat 2–4 steps if a good solution is not found.

The parameters of the GA's are given in Table 2. *GAmain* run 30 times for every *GAfit* generation; totally, the GAs produced 1200 generations.

5 Simulation results

The transition of the weights of the fitness function in the GA is shown in Figure 2.

The best trajectory has been obtained in the 71-th generation under the condition of $w_0 = 0.29, w_1 = 0.194, w_2 = 0.215$, and $w_3 = 0.30$, where w_i is the weight for i-th evaluation, and it is shown in Figure 3.

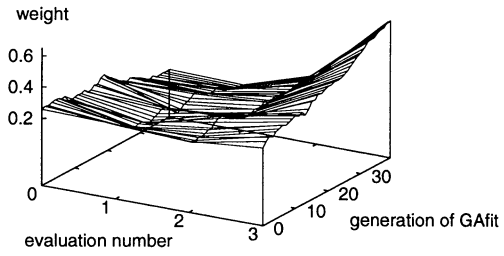


Figure 2. Transition of the fitness function.

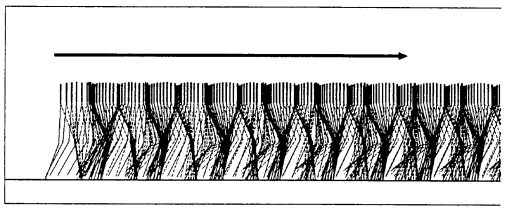


Figure 3. The best trajectory produced by the co-evolutionary GA.

Figure 4 presents the trajectory produced by an ordinary GA with the parameters assigned by an experienced human designer as follows: *population* = 300, *generation* = 200, and $w_i = 0.25$, for $i = 0, 1, 2, 3$.

Through an analysis of the results of the co-evolution, it has been found that decreasing the importance of *max step*(w_1) and increasing the importance of *summation of angles*(w_3) allows for generating the creatures which demonstrate the ability to walk far.

6 Concluding remarks

Designing a walking generator for a biped robot is a difficult task that requires creative thinking. This paper presented an attempt to model the cognitive processes underlying human creativity while developing a walking pattern generator. It has been shown that the interaction of divergent and convergent thinking or, in other words, design synthesis and analysis in the creative process can be computationally modeled in the framework of GA-based evolution defined on a design space. To model the transformation of the design space throughout the design process, a co-evolutionary scheme of the interaction of two GA's has been proposed. By applying this scheme, it becomes possible to model the primary cognitive creative processes and support human *impossibilist* as well as *improbabilist* creativity. A case study has been made that demonstrated the applicability of the proposed approach to the problem of designing walking pattern

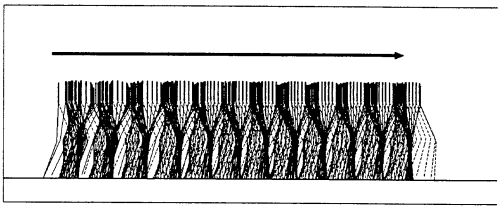


Figure 4. The trajectory produced by an ordinary GA.

generators. The developed software does not remove the need for the human designer, but it supports human creative thinking by enhancing the designer's analytical (through the application of the single GA) and synthetic (through the generation of new design spaces with the co-evolutionary algorithm) abilities.

Our future plans include the development of a general purpose simulation tool to support and enhance human creativity in designing biped robots.

This work has been made within the "Methodology of Emergent Synthesis" research project in the Program "Research for the future" of the Japan Society for the Promotion of Science (grant No 96P00702).

References

[1] "Engineering Design and Creativity.WDK 24", E.W.Eder(ed.), Edition HEURIST, 1996

[2] Kryssanov V. V., Tamaki H., Kitamura S. "Understanding Design Fundamentals: How Synthesis and Analysis Drive Creativity, Resulting in Emergence", *In Proceeding of IWES'99, the International Workshop on Emergent Synthesis*, Kobe University, 1999

[3] Matsuoka K., "Mechanisms of Frequency and Pattern Control in the Neural Rhythm Generators", *Biol. Cybern.*,Vol. 56, pp.345–353, 1987

[4] Inamoto T., "Generating a biped walking trajectory with GA", Graduation Thesis, Kobe University, 1999

[5] Nakamoto Y., "Acquiring a biped walk by applying Reinforcement Learing to construct an ANN", Graduation Thesis, Kobe University, 1996

[6] Wallas G., "The art of thought", Harcourt.NY Brace, 1926

[7] Goguen J., "An Introduction to Algebraic Semiotics, with Applications to User Interface Design",C.Nahaniv(ed.) *Computation for Metaphor, Analogy and Agents*, Springer Lecture Notes in Artificial Intelligence, Vol.1562, pp.242–291, 1999

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

HOME MEDICAL SUPPORT SYSTEM BY BILATERAL TELECOMMICATION BETWEEN PATIENTS AND DOCTORS

H.Wakamatsu*, S.Honma*, X.Zhang* and H.Yamamoto**

*Faculty of Medicine, Tokyo Medical and Dental University
1-5-45 Yushima Bunkyo-ku, Tokyo 113-8519, Japan
**Faculty of Engineering, Ashikaga Institute of Technology
268-1 Ohmaecho Ashikaga Tochigi 326-0845, Japan

ABSTRACT Three basic systems of telemedicine with bilateral communication using multimedia are applied to the medical and health care. They are supported by the technology developed in robotics and virtual reality. The first one is a home emergency rescue system for the patients of respiratory disorder. The second one is the relief system of senile elderly people from their possible outdoor danger using appropriate signals transmitted by telecommunication line. The third one is an automatic examination system of urinary components at home, which are stored for medical care. These teleoperational care systems based on the active utilization of multimedia infrastructure will become the facilities for our aging society in the coming new century. Various techniques are requested for the proposed systems, in which essential are hardware manipulating systems of 3D virtual devices and objects with their physical properties in use of telecommunication From this viewpoint, teleoperational force display system is developed which can provide an operator the well attendance to the realistic operation with a smooth reactive feeling dependent on the physical properties of the object to be operated

Keywords: Telemedicine, Bilateral communication, Virtual reality, Virtual object, Teleoperational force display system

1. INTRODUCTION

The necessity of telemedicine has been talked about lots for more than 30 years on demand of the convenience and economic effect in remote regions in our country. Meanwhile, the high density transfer of medical examination data of patients has been economically realized owing to the development of techniques of telecommunication. Thus, visual information has been easily provided which must be one of the most substantial techniques in the interactive communication [1-3]. Most of the concept of telemedicine applied to the practical use are however for the diagnosis or for the communication among medical experts; i.e. the utilization of medical information has been unilaterally considered only from the convenience of the people engaged in the medicine rather than from the demand of patients. Who requires medical care is, of course, not a medical staff but a patient that asks the aid in case of emergency.

Thus, it is desirable to send signals for help from the patient's side and to acknowledge the patient's demand by medical staffs. In that case, it is inevitable not only to transmit high quality of image for observation of the states of patients but also to make appropriate decision with instruction to the patients. Hereby are introduced the bilateral communication systems between patients and medical staffs in a definite regional group, which enable the former to ask the latter for direct help and also the latter to make physical diagnosis of the former with appropriate instruction.

In the present study, the basic three telemedical systems are first explained [4-6] and combined to teleoperational system in order to realize more attendance to the reality for the manipulation and processing of medical care [7-10].

2. MEDICAL SUPPRT SYSTEMS WITH BILATERAL TELECOMMUNICATION

2.1 Home Assist-Respirator in Emergency

There increase various troubles of respiration with the aging of the elderly. The portable practical respirator has been developed, by which alveolar CO_2-concentration and peripheral arterial O_2-saturation are non-invasively controlled within their desirable range of value by the method giving human-friendly ventilation. In practice, adaptive and/or fuzzy control theories have been applied to the control of respiration, which lead us to overcoming the difficulty and laborious works resulting from the different clinical attitudes to cope with the difference of individual patients. The asthmatic patients cannot in general rely on the visiting nurses and doctors in case of a sudden attack of asthma. Control system of respiration at a distance is developed for the emergency help of the elderly of respiratory disorder, especially the asthmatic patients as an application of the developed respirator. On one side, patients can directly ask doctors for medical aid. On the other side, doctors directly can deliver them medical instructions. The switch is automatically on, when a suffering patient wears emergency air-mask of assist-respirator on the mouth. Then, the contained computer starts with its automatic switching-on mechanism and makes him inspire the air mixed with oxygen and cough and cardiac remedy as prescribed by doctors, only just wearing the mask by himself alone at home. Not only medical effect can be expected in emergency to cope with sudden asthmatic attack, but also the help and rescue

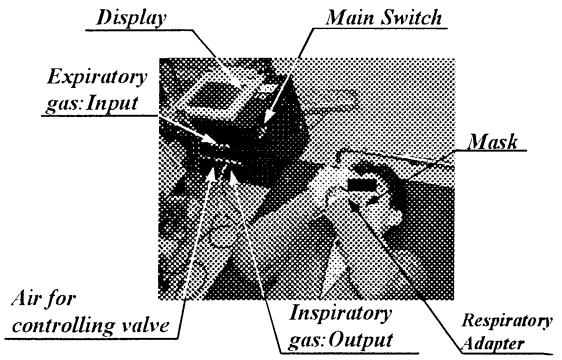


Fig.1 A patient using a developed home respirator.

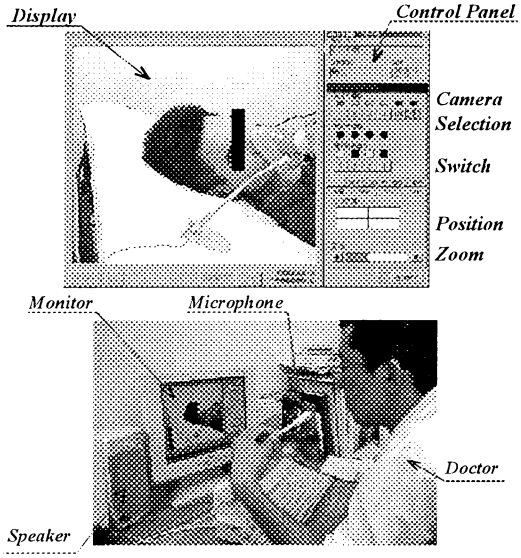


Fig.2 The communication between a patient and a doctor.

can be asked automatically with the aid of computer. It is remarked that this assist-respirator itself has been designed on the basis of the experience of expert doctors and control theory so that the patients can easily inspire and expire in artificial respiration not only in case of emergency such as asthma but also in chronically muscular dystrophy without any special assistance of medical staffs. It is practical not only for the assisted-respiration to the patients suffering of hard respiration but also for the benefit of QOL outdoors with its setting up on wheelchairs. That is, it is useful for patients of respiratory disorder, not only at home but also on their

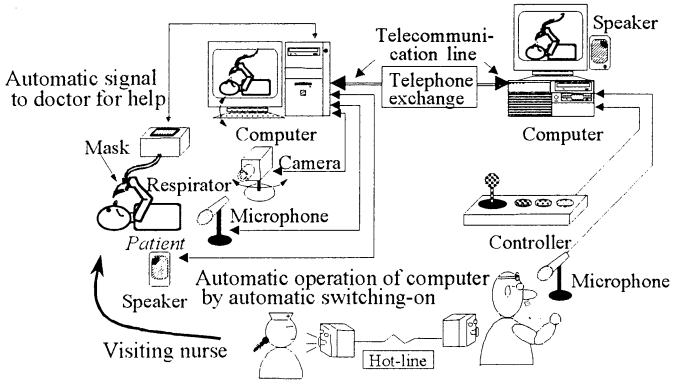


Fig.3 Concept of bilateral telemedicine for the elderly living alone at home

outgoing by its mobile use for their QOL improvement, as it is being reformed to reduce its size and weight for the convenience to carry. It is emphasized that this system can be applied to the first aid in the absence of medical staffs, in emergency in an airplane, in a train, in an ambulance, etc. Within a minute from automatic start of the computer after wearing the mask, the system is first available as a conventional assist-respirator. Then, it makes an automatic dial to the registered doctors in a definite region where the patients live and the doctors acknowledge their signal for help by sound and light. On having received emergency information, the doctors use the same telecommunication line to send the operational signal back to control the position and attitude of small video-camera to make the image information clear by using a stick as illustrated in Fig3. That enables the doctors to give a patient emergency treatment or instruction observing his face and behavior. If the state and condition of the patient improve or become tranquil during this period, instruction by sound through microphone is thought enough. The places of the dispatch and the sender of signals for help can be confirmed by sound and image. Information of the case history and behavioral state of the patient are automatically displayed on the CRT monitor for the doctors. If the doctor regards that further care and treatment are necessary from the image and voice information of the patient, he can ask the helpers and/or nurses from their stations to visit the patient at home or he visits the patient at home by himself. It is thereby of course possible for patients to ask doctors to dispatch the visit of above medical staffs on their direct demand.

2.2 System for Controlling Senile Elderly Patients

The electrical safeguard system has been developed for senile dementia patients who wander without definite purpose because of declined mental capability, while retaining their physical ability. The system has been applied to more than 200 special nursing homes for more than 13 years. This system ensures their relief from danger at their unexpected outgoing by informing their behavior, offering them an environment of free intercourse with their visitors. Its application at special nursing homes ensured that the wandering elderly men could be brought safely back indoors after going outdoors alone. In addition, their mental health improved because they could behave themselves at will in an atmosphere of freedom. It was also observed that helpers were considerably released from physical and psychological burdens so that they could devote themselves to the care of elderly men, thereby improving their patients' comfort. Using data

acquisition function of the safeguard system, relief of senile elderly people by helpers can be informed through telecommunication line by remote control of the computer which belongs to the system. Much care is requested at home for the safety of the wandering senile elderly patients in comparison with the ones at special nursing homes, because they have higher probabilities to encounter outdoor dangers. Thus, the system has been developed as a small all-built-in-one type apparatus suitable for the home use, in which wide space is not always expected for its setting up. In this case, necessary apparatus is easily set up near the entrance without any special construction of transmission line. During the night, radio pagers and PHSs are used informing alarm to plural number of helpers by their vibration in order that the others may not be disturbed to sleep. Utilizing the network of electronic safeguard systems by telecommunication line, operational data of the systems, meteorological data and vital signs are monitored from the distance. The elderly may well only wear talisman type, wristwatch type and embedded type transmitters. The receivers give the helpers by melody chimes and color lamps the sound and light warnings, which are simultaneously transmitted to the distant place by portable alarm using distribution line. The helpers take trace and care of the patients during their housekeeping by a FM-radio and a cordless telephone, and can ask some neighbors to take care of the elderly even on the way to their shopping and outgoing by radio pagers or PHSs through telecommunication line.

2.3 Automatic Examination of Urinary Components for Individual Health Care at Home

There have been gradually increased the social requirement of automatic check and care of healthy individuals in everyday life. Automatic examination system of urinary components with automatic identification of individuals is developed for the everyday health care at home, especially for the early discovery of disease. This system is only for a small group or a family unit, because the psychological restriction can be avoided only within an intimate group of persons. Under the condition that the subjects are unaware of the ongoing examination, their easily obtainable physical information is picked up and analyzed automatically. The personal identification is known by fingerprints, the fingers of which push the button to open the door of rest room electrically, taking into account everyday urination. Urinary components such as protein, glucose, ketone, nitrite, leukocyte and occult blood are examined by the color change of the stripe-type test papers, which are supplied and wasted automatically. The examination result is automatically displayed and stored in a disk of computer. If something different or abnormal from usual or normal states, its warning is directly given to the subjects. The examination data are automatically transmitted also to medical staffs by telecommunication line in such a case.

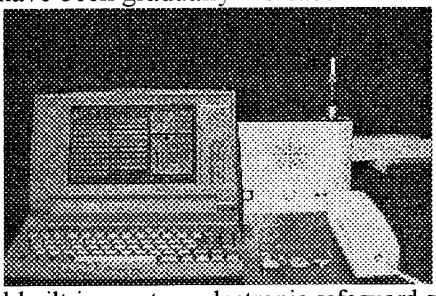


Fig.4 All-built-in-one type electronic safeguard system with computer and portable transmitter.

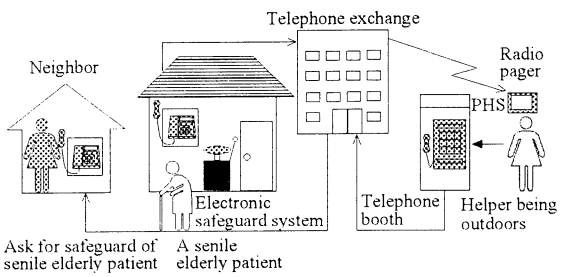


Fig.5 Protection of wandering senile elderly by helpers outdoors.

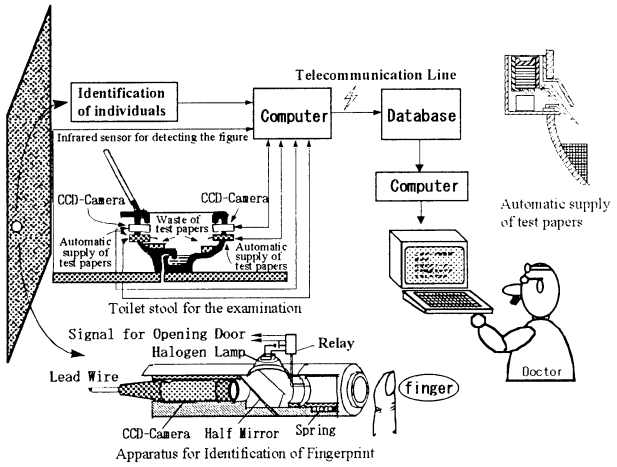


Fig.6 Automatic home examination system of urinary components with automatic identification of individuals

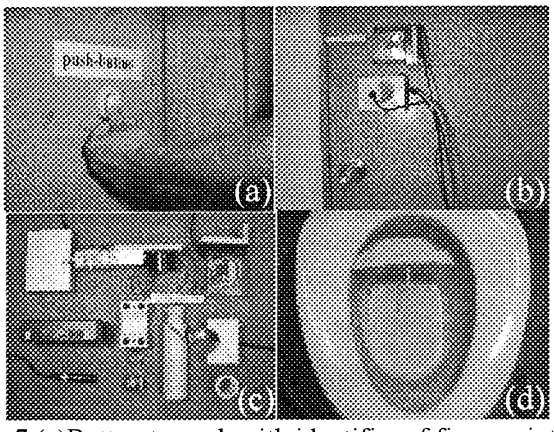


Fig.7 (a)Button to push with identifier of fingerprints, (b) Electronic door lock, (c)Parts of the system and (d) Toilet stool with supply of test papers.

3. TELEOPERATIONAL HOME MEDICAL CARE SYSTEM

3.1 Teleoperational Force Display System

It is important to assure doctors to operate the systems at a distance and also to attend to the reality when they are consulted by patients. Two same actual devices are developed, by which force feeling and movements of the fingers are recognized. One is set on the doctor's side for the manipulation of stereoscopic 3D image of something on the patient's side. The force display system is constructed, in which the actual device is manipulated, and the virtual device and object are simultaneously seen on the screen in stereo-view using binocular parallax. On the patient's side, there should exist the same actual device with the actual object. The virtual object is given by combinations of tetra-polyhedron viscoelastic element models with mass on their apices. Actually, the reactive force is given to the finger on the operation of the object according to its physical property through telecommunication line [8-10].

3.2 Attendance to the Reality

Attendance to the reality is based on the virtual reality techniques. The function of the actual operating device is controlled on the operation of stereoscopic image object by the actual manipulating device from a distant place. The movement of the virtual device is reflective of the actual devices on both sides. The computer drives the electromagnetic brakes mounted on the actual operating device in order to give a reactive force to the operator according to the position and attitude of the virtual device and object. Then, the operator can, in a real time, realize the action through his actual device with the virtual device and object, obtaining the response from the actual object to be operated. This kind of operation is essential for the actual medical diagnosis, treatment and action when the telemedicine is legally admitted. They will be, for example, the support technique for the operation of jet nozzle of liquid medicine, for the control of the pressure of air in the previously mentioned assist-respirator and for the appropriate diagnosis of patient's state by questioning, tapping , etc.

4. CONCLUSIONS

The three basic works of telemedicine have been introduced in this study. The concerning systems have been developed mainly for the living alone at home, utilizing multimedia infrastructure in order to reduce increasing medical expenses in our aging society. Because of various legal restrictions of our society at present, the proposed systems cannot be directly applied to the clinical medicine. However, they are at least used for the medical aid and calling help in emergency. The first one is put to practical use for the rescue of emergency respiratory disorder in the absence of medical staffs. The second one is applicable to the home care of the senile elderly, the feasibility of which has been confirmed in practical use. The third one is used for rough self-screening because the concentrations of urinary components are measured only by color change of the test papers. The basic teleoperational force display system has also been introducedusing our developed hardware devices to manipulate actual object with the help of virtual device and object. The virtual device is associated with the actual devices as master-slave manipulators. That ensures operators the well understandings of the positional relation between actual device and actual object by the visual recognition of a hand movement, which leads to the way of their application to the actual medical manipulation. There exist stereoscopic virtual device and object in and in front of CRT, whereby operating feeling and satisfactory attendance to the reality are artificially given to the operators of actual device. It is hereby remarked that the teleoperational force display can contribute to the telemedicine in the near future, since the basic techniques to transmit the information about patients with devices and objects have been established, which are used for operation of actual device by medical staffs at a distance.

REFERENCES

1. H.Wakamatsu, T.Imai and K.Okada: Artificial realization of reactive force feeling on stereoscopic cutting of image of multi-layer sphere. Proc. 2nd Int. Conf. Image Process., 552-556 (1992).
2. H.Wakamatsu and T.Imai: Development of stereoscopic image cutting system with reactive force feeling. Trans. Inst. Electr. Eng. Japan, 113-C, 8, 627-634 (1993).
3. H.Wakamatsu: Operational systems of stereoscopic cutting 3D virtual objects with reactive feeling. Proc. IFAC 13th World Congr., Biomed. Control, 409-414(1996).
4. H.Wakamatsu, S. Honma and X.Zhang: Development and characteristics of small home use respirator for the elderly at home. Jap. J. Appl. Physiol. 199-205 (1999).
5. H.Wakamatsu and H.Yamamoto: Electronic identification and safeguard system network of senile elderly patients. Proc. Workshop. Jap .Med. Electron. Biol. Eng.158 (1994).
6. H.Wakamatsu and X.Zhang: Development of examination system of urinary components with automatic supply of test paper. Proc. 34th Congr. Jap. Med. Electron. Biol. Eng. 460 (1995).
7. H.Wakamatsu, X.Zhang and S.Honma: Teleoperational force display system in manipulation of virtual object using scissors-type cutting device. Proc. Asia-Pacific Conf. control Meas., 312-316 (1998).
8. H.Wakamatsu and S.Honma: Home medical support system in emergency with interactive telecommunication, Proc. 6th Asia/ Oceania Region. Congr. Gerontrol., 243 (1999).
9. H.Wakamatsu, X.Zhang and S.Honma: Teleoperational force display system of virtual object using scissors-type cutting device. Trans. Inst. Electr. Eng. Japan,119-C, 10, 1126-1132 (1999).
10. S.Honma, H.Wakamatsu and X.Zhang: Mechanics and manipulation of virtual objects represented by viscoelastic model. Trans. Inst. Electr. Eng. Japan, 119-C, 12, in press.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Modular Neural Network Architecture for Inverse Modeling of Mobile Robots

S. Yamaguchi and H. Itakura
Department of Computer Science
Chiba Institute of Technology
2-17-1, Tsudanuma, Narashino-shi, Chiba, Japan 275-0016
E-mail: yamag@ita.cs.it-chiba.ac.jp

Abstract

A new modular neural network architecture and its learning algorithm for inverse modeling of mobile robots are proposed in this paper. In our architecture a complex task to be executed by robots is divided into some small simple tasks and each task is assigned to small network modules. The output value of the modular neural network used as a command of the robots is the weighted sum of the output values of network modules. By dividing the whole complex task into small simple tasks, the networks are able to be trained easily.

The proposed modular neural network architecture and its learning algorithm are applied to a task of pushing a box by two robots. It is confirmed through real robot experiments that the architecture can carry out robot controller skillfully.

key words: modular neural network, inverse modeling, mobile robots

1 Introduction

Recently, autonomous mobile robots are actively utilized as very good application examples in various research fields such as artificial intelligence, evolutionary algorithm, artificial life and so on [1, 2]. The problem of obtaining an effective mobile robot controller by using a neural network is regarded as the inverse modeling of the robot. The feedback error learning [3] is one of useful learning algorithms for obtaining an inverse model by neural networks. When the feedback error learning is applied to some system, the feedback controller of the system is required in the learning scheme. If the tasks and environments of mobile robots are simple, then the feedback controllers of the robots are obtained easily. However, if the tasks and/or environments are complex, then the controllers of the robots are obtained not so easily, because the relationship between the sensor values and the robot actions can not be represented so easily. In such a case, the feedback error learning can not be effectively applied to this kind of task.

In this paper, a modular neural network architecture and its learning algorithm are proposed for the inverse modeling of the mobile robot. In the learning algorithm, a complex task is divided into some simple and small tasks, and each simple task is assigned to each network module which is included in the proposed modular neural network. Each module has its own feedback controller for solving the simple problem assigned to itself. These modules are called feedback error learning (FBEL) block. By dividing the complex problem into some small problems with simple structure, the feedback controller of each problem can be obtained easily. Hence, the network module can be trained by the feedback error learning. The command to the robot is the weighted sum of the output values of each module. The weights for each module are obtained from a network module included in the proposed modular network. This network module is called the gating network module. The feedback error learning modules and the gating network modules are trained together in the learning scheme. The reinforcement learning method is applied to the training of our modular neural network. The robot actions, which carry out the task correctly, are given some reward, and the network modules are trained in order to maximize the total reward by using the feedback controller.

2 Network architecture

In robot control, the robot receives a command from the controller and makes an action in the environments. After that, the robot obtains some results from environments as sensor values which are able to be considered as the output value of the robot. The controller of the robot receives the sensor values of the robot and target sensor values and generates com-

mands to the robot. The target sensor values are the desired sensor values as results of the robot action from the command. Hence, obtaining a robot controller is regarded as inverse modeling of a system. In our proposed method, the neural network and the feedback error learning algorithm[3] are utilized for obtaining an inverse model of robots.

In order to obtain an excellent controller of the mobile robot which carries out some task (for example, pushing a large box by two robots) by the feedback error learning, a feedback controller of the robots is required. If the task is not so complex, then the feedback controller can be obtained easily. The task is, however, complex and then it is difficult to obtain an excellent feedback controller for the mobile robot. In case of a problem of pushing a box in unknown environments, the controller has to decide how to search the box, how to push the box, and how to avoid the wall. If the number of the robots is larger than one, the task will be so more complex that we would not be able to obtain a feedback controller so easily. Some kinds of complex problems, however, are able to be divided into some simple problems and it is not so difficult to obtain a feedback controller for each simple problem.

In our proposed method, the modular-type neural network architecture is applied to this kind of complex problem. At first, the problem is divided into some simple problem and each simple problem is assigned to one neural network module which is trained by the feedback error learning. These neural network modules are called the feedback error learning (FBEL) blocks. After that, the output values of each module are mixed by another neural network which is called the gating network. The resultant structure of our network is shown in Fig. 1. Each FBEL block is consisted of a feedback controller and a neural network which is trained by the feedback error learning.

The input values of the neural network correspond to the sensor values $\boldsymbol{s}$ of the robot and the output values are associated with the command $\boldsymbol{v}_i$ to the driving system of the robot to solve a set of small problems.

The output value of the modular neural network used as the command to the robot is the weighted sum of the network modules so that it is calculated by

$$\boldsymbol{v} = \sum_{i=1}^{n} g_i \boldsymbol{v}_i \tag{1}$$

where g_i is the output value of the ith gating network, $\boldsymbol{v}_i$ is the output value of the ith neural network module and n is the number of the network modules.

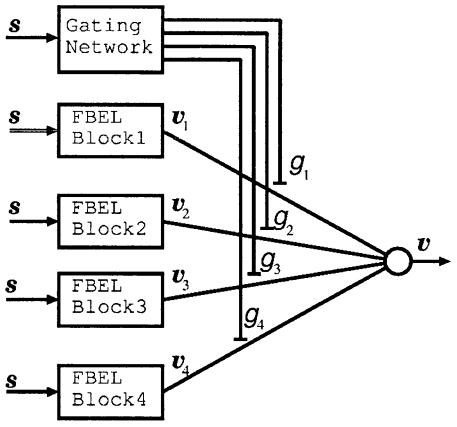


Figure 1: Modular neural network model.

3 Learning Algorithm

The learning algorithm for the previous modular neural network is descried in this section. The learning algorithm of the gating network is based upon the modular neural network architecture learning algorithm[4] and the algorithm of the FBEL blocks is based upon the feedback error learning algorithm [3].

The output value g_i of the gating network is calculated from

$$g_i = \frac{\exp(\xi_i)}{\sum_{j=1}^{n} \exp(\xi_j)} \tag{2}$$

$$\xi_i = \sum_{j=1}^{n} u_{i,j} s_j \tag{3}$$

where $u_{i,j}$'s are the value of the synaptic weight connecting from the jth input unit to the ith output unit of the gating network, s_j is the input value to the gating network, and n is the number of the output units.

Before training of network modules in our modular neural network, training data need to be prepared. To prepare a set of training data, the robot tries to carry out the task, and then the behaviors of the robot are evaluated as 'good', 'bad' or 'neutral'. The data with 'bad' are thrown away. The input sensor value $\boldsymbol{s}(t)$ and command $\boldsymbol{v}(t)$ evaluated as 'neutral' behavior are stored in memory while 'good' behavior appears. When data with 'good' appear, the network modules are trained by data with 'good' and the weighted stored data. An example of stored data is shown in Table 1 where γ is the weight having a positive value less than one for data with 'neutral'.

If the behavior is good, the gating network is

Table 1: An example of stored data

sensor	command	evaluation	weight
$\boldsymbol{s}(1)$	$\boldsymbol{v}(1)$	bad	0
$\boldsymbol{s}(2)$	$\boldsymbol{v}(2)$	neutral	γ^{p-2}
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\boldsymbol{s}(p-2)$	$\boldsymbol{v}(p-2)$	neutral	γ^2
$\boldsymbol{s}(p-1)$	$\boldsymbol{v}(p-1)$	neutral	γ
$\boldsymbol{s}(p)$	$\boldsymbol{v}(p)$	good	1

trained by

$$\Delta u_{i,j} = \rho \sum_{p=0}^{P} \gamma^p (h_i(t-p) - g_i(t-p)) s_j(t-p) \quad (4)$$

where $\Delta u_{i,j}$ is the updating value of synaptic weight in the gating network connecting the jth input unit to the ith output unit, $\boldsymbol{s}_j(t-p)$ are the data which are evaluated as 'neutral' and stored in memory in the past evaluation process, P is the number of data stored in memory, ρ is a small positive number and h_i is calculated as follows:

$$h_i(t) = \frac{g_i(t) \exp(-\|\boldsymbol{v}(t) - \boldsymbol{v}_i(t)\|^2/2)}{\sum_{j=0}^{n} g_j(t) \exp(-\|\boldsymbol{v}(t) - \boldsymbol{v}_j(t)\|^2/2)} \quad (5)$$

Then the neural network of each FBEL block is trained through

$$\Delta \boldsymbol{w}_i = \sum_{p=1}^{P} \gamma^p h_i \boldsymbol{v}_i^{FB}(t-p) \boldsymbol{s}(t-p) \quad (6)$$

where $\Delta \boldsymbol{w}_i$ is the updating value of synaptic weights of the ith module's neural network, P is the number of $\boldsymbol{v}^{FB}$. After updating the synaptic weights, the memory is cleared. After the learning, the controller is able to obtain appropriate commands from the sensor values.

4 Simulation

4.1 Khepera robot

The Khepera robot [2] is employed in our experiments to examine the algorithm proposed in this paper. This type of robot is often applied to a lot of experiments of autonomous mobile robots through using neural networks, genetic algorithms, evolutional algorithms, etc.[2, 5].

The Khepera is round in shape and measures 55mm in diameter. It has two wheels which are controlled to be driven by two DC motors independently one another, and are able to rotate in both directions. With changing each wheel speed, the robot can move freely around in the two dimensional space.

The eight infra-red proximity sensors (IR sensors) are equipped around the Khepera body (the six sensors are on front side and the remaining two sensors are on rear side). Each sensor outputs the values normalized from 0 to 1023 in integer responding to the distance from the Khepera to obstacles. The output value 0 means that there is no obstacle near to the sensor. With the aid of the output values from these sensors, the Khepera can recognize its own environments, so that the robot can understand the local environments only.

In order to recognize the objects and/or the wall, a vision sensor module is attached to the Khepera. This sensor consists of a 1D-array of 64 photoreceptors which provide a linear image composed of 64 pixels of 256 gray-levels each, subtending a view-angle of 36 degree.

4.2 Simulation results

To confirm that our algorithm can succeed in accomplishing the robot controller learning, the algorithm is applied to learning a controller for pushing a box by several robots. In our experiments, instead of real robots, a computer program simulating the Khepera is constructed. The range from 0 to 1023 of the IR sensor values of the Khepera is normalized to be converted from 0 to 7 in integer in the simulator. The Khepera has a vision sensor. The normalized range of the vision sensor values is from 0 to 256, but it is converted simply to 0 or 1 in the simulator. From these vision sensor values, the input value of the sensor is calculated by

$$C = \sum_{i=0}^{64} (i - 32) K_i \quad (7)$$

$$R = K_{\max} - K_{\min} \quad (8)$$

$$K_{\max} = \max\{i | K_i \neq 0\}$$

$$K_{\min} = \min\{i | K_i \neq 0\}$$

where K_i ($i = 1, \cdots, 64$) is the value of the ith vision sensor unit, and C and R are used as the input value of the feedback controller.

The environment in which the Kheperas and the box move around is of 595mm × 840mm square. This environment is surrounded by the white wall. The box in the environment is 300mm in width, 40mm in

depth, and 40mm in height and has two black posts being 14mm in diameter and 50mm in height. The color of the wall is so white that the vision sensor can recognize the black bars on the box. The IR sensors, however, can not recognize the box and the wall; it can recognize only whether some obstacles are near or far.

In our experiments, the four modules are utilized, so that the modular neural network has four FBEL blocks and one gating network module. The first is for searching the box, the second module is for pushing the box, the third module is for moving forward the robot, and the forth module is for waiting the other robot. The number of output units of the gating network is four.

The evaluation of the Khepera's behavior is as follows: If the Khepera finds the box, the post is in front of the Khepera, and the Khepera moves forward, then the behavior is evaluated as 'good'. If the Khepera does not find the post, some obstacles are in front of the Khepera, and the Khepera moves forward, then the behavior is evaluated as 'bad'. The other behaviors are evaluated as 'neutral'.

Figure 2 is an example of the results of the robot behavior, after five hundred training. In the training of the network modules, a simulator of the Khepera and the environment are used. After that, the trained modular neural network was is applied to the real Khepera and the environment.

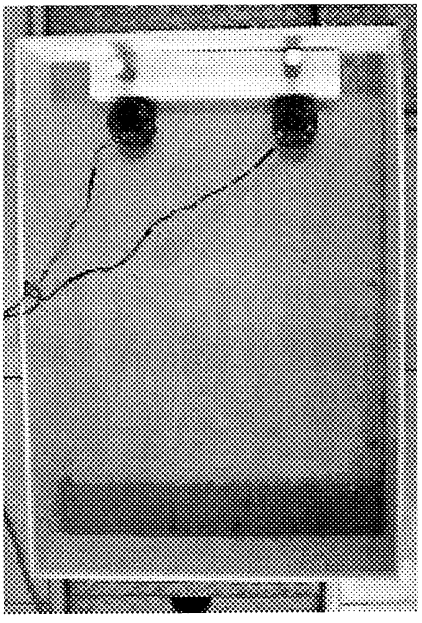


Figure 2: An example of the experiment results

5 Conclusion

This paper has proposed a modular neural network architecture based on the feedback error learning for an inverse modeling of mobile robots. In our architecture, each module consists of a feedback controller and a neural network. These modules solve derived simple problems and, by using the gating network, the controller obtains the output values for solving complex problems. Because each module is assigned to a simple problem, the feedback controller is able to be obtained easily. The command to the robot is calculated as the weighted sum of the output values of FBEL blocks, and then the weight is calculated by a network module named gating network which is trained with FBEL block. Our proposed method is able to be applied to the case in which the complex task is divided into some small simple tasks.

To confirm that our architecture performs properly, several results of real robot experiments have been shown. We can conclude from these results that the proposed modular neural network architecture gives good mobile robot controllers which solve a complex control problem.

References

[1] Brooks, R., "A Robust Layered Control System for a Mobile Robot", IEEE Journal of Robotics and Automation, Vol. RA-2, No. 1, pp. 14–23, 1986.

[2] Donnart, J. Y. and Meyer, J. A., "Learning Reactive and Planning Rules in a Motivationally Autonomous Animate", IEEE Trans. on Systems, Man and Cybernetics, Vol. 26, No. 3, pp. 381–395, 1996.

[3] Kawato, M., Furukawa, K. and Suzuki, R., "A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movement", Biological Cybernetics, Vol. 57, pp. 169–185, 1987.

[4] Jacobs, R. A. and Jordan, M., "Learning Piecewise Control Strategies in a Modular Neural Network Architecture", IEEE Trans. on Systems, Man and Cybernetics, Vol. 23, No. 2, pp. 337–345, 1993.

[5] Floreano, D. and Mondada, F., "Evolution of Homing Navigation in a Real Mobile Robot", IEEE Trans. on Systems, Man and Cybernetics, Vol. 26, No. 3, pp. 396–407, 1996.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

DESIGN SPACE TRADEOFF IN VLSI IMPLEMENTATIONS OF MIXED-SIGNAL NEURO-FUZZY PROCESSORS

J. Madrenas, E. Alarcón, J. Cosp, J.M. Moreno, A. Poveda and J. Cabestany

Department of Electronic Engineering, Technical University of Catalunya,
Jordi Girona 1 i 3, Mòdul C4 Campus Nord UPC, 08034 Barcelona, Spain
Phone: +34 93 401 67 47, Fax: +34 93 401 67 56, E-mail: madrenas@eel.upc.es

Key words: Sequential processing Mixed-signal, Neural networks, Fuzzy controllers, Design space, VLSI

ABSTRACT

Mixed-signal VLSI design provides a tradeoff between the fast and compact but fixed analog implementations of neuro/fuzzy feedforward algorithms and the programmable but area and power consuming digital counterparts. In this paper, a sequentiality study of such systems is performed. Basic blocks for sequential mixed-signal neural and fuzzy computing are proposed, and two sequential example processors are described. Feedback from the designed processors and subcircuits allowed considering the technology constraints for analysis and extension to different sequentiality degrees.

1. INTRODUCTION

Efficiency in terms of silicon area occupancy versus computation performance (speed) is fundamental in autonomous intelligent systems. For perception tasks, such systems take advantage of several proposed neural and fuzzy models. In real-time operation, hardware implementation of the required models can be most suitable. However, for a given application, the hardware resources could be adapted to the specific performance requirements in order to reduce silicon and power cost. In this paper we analyze mixed signal sequential architectures that enable this tradeoff.

Hardware implementations of neural networks and fuzzy controllers by means of analog processing exhibit compact, fast and low power characteristics in front of digital realizations. This is achieved at the expense of low resolution, sensitivity to noise, temperature and non-idealities [1-3].

An important weak point in analog implementations is the lack of suitable memory elements. However, in the last years several dynamic and non-volatile analog memory circuits have been developed. This enables the implementation of compact analog architectures.

Although precision requirements for the learning phase can be high, for the recall phase they are less than 8 equivalent bits in most neural applications [4]. Therefore, proper analog circuits can clearly perform this phase. In addition, learning can be done by means of chip-in-the-loop techniques.

Usually, analog implementations are fully parallel one-to-one mappings of the processing network. This results in high complexity of connections between processing elements as a consequence of the so-called curse of dimensionality. Despite the compact analog implementations, for real-world applications the number of synapses and processing units usually becomes too high to justify a fully parallel implementation, especially when speed can be reduced and still match the application needs.

In contrast, hybrid parallel/sequential architecture can match the analog processor cost, speed and power consumption to the application constraints. Furthermore, in general, sequentialization exhibits flexibility to emulate a different number of network structures (e.g. evolutive networks) and models with much better efficiency than a fully parallel architecture.

As far as analog sequential processing is concerned, we can take advantage of several developments on switched current circuits. These techniques perform current-mode, discrete-time processing [5], using the MOS gate capacitor for temporary charge storage.

Being analog dynamic memories available, sequential processing schemes can be envisaged. A digital block becomes necessary to control the sequential processing, so such systems require mixed-signal design techniques. However, in contrast to most mixed-signal circuits, a distinct feature of the proposed circuits is that main processing is performed at the analog part. Although several mixed-signal sequential realizations have been reported, e.g. [6-9], to our knowledge no study has been done up to now on generalization of the area-delay tradeoff. In this paper, we consider two reported sequential realizations that cover feedforward algorithms (MLP, RBF and Sugeno-Takagi fuzzy controllers) for the *recall phase*. The common points and sequentialization possibilities are analyzed, and a consistent realization style that allows tuning sequentiality is proposed.

In section 2 we explore the area-delay design space of analog neural and fuzzy processors as a function of sequentiality degree. The VLSI implementation of two test processors is summarized in section 3. An example of cost-performance tradeoff is described in section 4. Finally, conclusions are discussed.

2. SEQUENTIALITY ANALYSIS

The trivial solution for an analog implementation of a neural or fuzzy system is one-to-one mapping of each synapse and activation function of processing elements onto physical resources. In this case, internal memory is only needed to store the system parameters.

At the opposite side lies a single processor emulating serially synapses and activation functions. In this case, in addition to parameters, memory is required to store temporary intermediate values, as happens in single-processor digital implementations. The equivalent mixed-signal elementary processor is shown in

Fig. 1, for a) Multi-Layer Perceptron (MLP), b) Fuzzy Knowledge-Based Controllers (FKBC) and c) Radial Basis Functions (RBF). Aside of the nonlinear activation function and distance/minimum calculations, the multiplier-accumulator is a ubiquitous element. It uses a temporary storage element (Analog Transfer Cell, ATC). The digital controller and parameter storage elements are not shown.

In between the extreme solutions, any combination of processors executing in parallel, each one emulating several processing elements, is possible. The following subsections comment on two reported architectures of this kind.

2.1 MLP-like emulation. In Fig. 1d, a synapse-parallel MLP emulating processor is shown [8,10,11]. For this kind of systems, either synapse or neuron parallelism can be applied. In the former, synapse aggregation takes place in space, while it takes place in time for the latter.

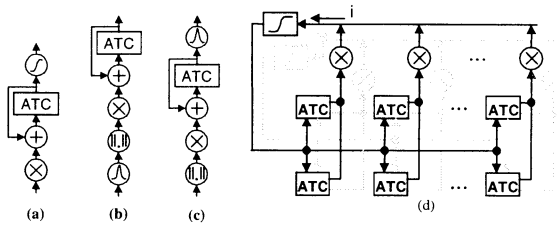


Fig. 1. a) MLP slice; b) fuzzy slice; c) RBF slice; d) MLP sequential synapse-parallel realization.

The processor operation principle is based on the sequential emulation of each neuron on a layer-by-layer basis. Let us consider only the black lines. Briefly explained, the input data vector is assumed to be stored in a first row of ATCs. At each clock cycle, all the synapses of one neuron and its activation function are computed. The activation outputs are sequentially stored in the second row of ATCs. After emulating all the neurons on the first neuron layer, the computed outputs can be used as inputs to emulate the next layer. The procedure is repeated until last layer is emulated. Two rows of n ATCs, n synapses and one activation function are enough to emulate a network of an arbitrary number of layers, each one consisting of up to n neurons.

The shadowed drawing in Fig. 1d shows the additional elements needed to add one degree of neuron parallelism to the processor. The area overhead is almost duplicated (only the ATCs stay the same), but also speed since two activation outputs are calculated at each time.

Comparing synapse and neuron parallelism, the former is more area efficient since the number of required activation functions reduces to one, while for each neuron to emulate in parallel an activation function circuit is needed.

Also, layer parallelism can be considered. As in the case of neuron parallelism, for each parallel layer an activation function is required. Furthermore, in this case additional ATCs to store intra-layer data are needed. In Table I, resource figures and execution cycles for different degrees of sequentiality schemes are shown, where s_s, s_n and s_l represent the synapse, neuron and layer sequentialization indexes, the maximum value representing fully sequential and 0 fully parallel. For simplicity, n stands for the number of synapses per unit and also units per layer. However, extension is straightforward. Index m represents the number of layers. The first row corresponds to a fully sequential system, the shadowed row represents the reported example synapse-parallel processor [10], and the last row corresponds to a fully parallel system. Control overhead is not considered.

	s_s	s_n	s_l	# p	# f.	#ATC	T
1	n	n	m	1	1	2n	$m \cdot n^2$
2	0	n	m	n	1	2n	$m \cdot n$
3	n	0	m	n	n	2n	$m \cdot n$
4	n	n	0	m	m	$2n \cdot m$	n^2
5	n	0	0	$n \cdot m$	$n \cdot m$	$2n \cdot m$	n
6	0	n	0	$n \cdot m$	m	$2n \cdot m$	n
7	0	0	m	n^2	n	2n	m
8	0	0	0	$n^2 \cdot m$	$n^2 \cdot m$	0	1

Table I. Resources and execution time function of sequentiality. # p: Multipliers; # f: Activation functions; T: Clock cycles.

2.2 Fuzzy/RBF-like emulation. The second sequential example architecture was reported in [9]. It is depicted in Fig. 2a, and, concerning hardware complexity, efficiently maps the Fuzzy Knowledge-Based Controller (FKBC) for the widely-accepted Takagi-Sugeno first-order model [12].
Again, the point of the sequential fuzzy controller is the capability to exchange speed for area and emulation flexibility. Table II summarizes the hardware complexity of the inference block (Fig. 2b) in terms of the degree of sequentiality. An n-dimensional input space, with a granularity of m membership functions per input dimension, and the associated m^n rules are considered as the general description of the operation of the fuzzy controller.

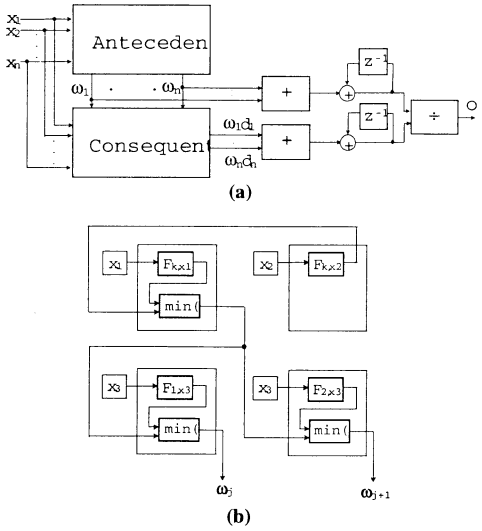


Fig 2. a) Diagram of the FKBC analog sequential architecture example. b) Inference organization.

Considering a designed testchip [13], which is a 3-input and 2-membership-functions per input FKBC, an architecture exhibiting a certain degree of sequentiality (complexity spread over time) and parallelism is considered (shaded row in Table II). In Fig. 2b, the configuration for the inference block with 3 inputs (x_1, x_2, x_3) and 2 membership functions are defined for each input. Concerning the consequent block, it consists of a weighted sum of products. Therefore it can be analyzed with the guidelines given for MLP implementations. The consequent block implements a 1st-order Sugeno model, very suitable for control applications [12]. The blocks labelled $F_{k,xi}$ in the figure implement the k-th membership function for the i-th input,

while the blocks labelled *min()* provide at their outputs the minimum of their two input signals. The membership functions for the two first inputs are calculated sequentially, while those of the remaining input are calculated in parallel. Thus, at each emulation step two fuzzy rules comprising the 3 inputs are yielded. As a consequence, the 8 rules of the system are calculated in 4 emulation cycles. In a general case of a fuzzy system with n inputs and m membership functions defined per input, the inference block would provide the complete set of fuzzy rules $M=m^n$ in m^{n-1} cycles. The number of membership functions required is $(n-1)+m$, as Table II shows.

s	# M.F.	# min.	T	# O
n	n	n-1	m^n	1
n-1	(n-1)+m	(n-2)+m	m^{n-1}	m
...	...	...	...	...
2	2+m·(n-2)	$1+\sum_{i=1}^{n-2} m^i$	m^2	m^{n-2}
1	1+m·(n-1)	$\sum_{i=1}^{n-1} m^i$	m^1	m^{n-1}
0	m·n	$\sum_{i=2}^{n} m^i$	1	m^n

Table II. Architecture complexity of the inference block as a function of the degree of sequentiality *s*. # M.F: Number of membership functions.
O: Number of fuzzy rules delivered in one cycle.

3. VLSI IMPLEMENTATION OF MIXED-SIGNAL SEQUENTIAL PROCESSORS

Two mixed-signal sequential test processors emulating the described MLP and FKBC models have been implemented in a full-custom testchip, using a 0.8μm CMOS technology. A photograph of thee manufactured chip is shown in Fig. 3. The chip is divided in four parts: (1) 8-bit 3-synapse MLP processor, (2) 8-rule/3-input FKBC processor, (3) Standard-cell digital control common for both processors and (4) Analog delay circuit for digital control signals to reduce switching noise during analog storage.

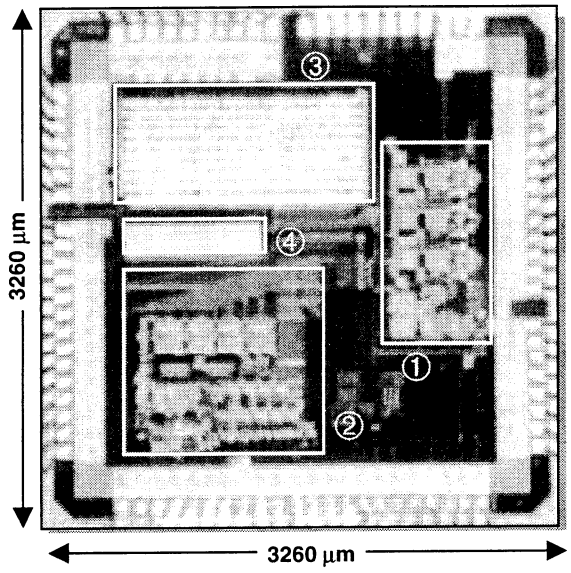


Fig. 3. Testchip including MLP and FKBC sequential processors.

Additional logic and resources were provided to have access to the basic elements of the processors, in order to be able to validate each one individually, so area could be reduced in a final design. Also, the flexible digital controller enables checking all the processor capabilities. It can also be significantly reduced.

Fig. 4 shows an enlarged photograph of the MLP processor. Its area occupancy is 0.79 mm^2, and it includes three synapse slices (number 1 in the picture), a sigmoid function (2) and an 8-bit scaling circuit (3). The synapse slice contains two ATCs (4) [14,15] and an 8-bit mixed digital-analog multiplier (5) [16-18], as well as control logic. The synapse-slice scheme allows a straightforward change of the number of synapses.

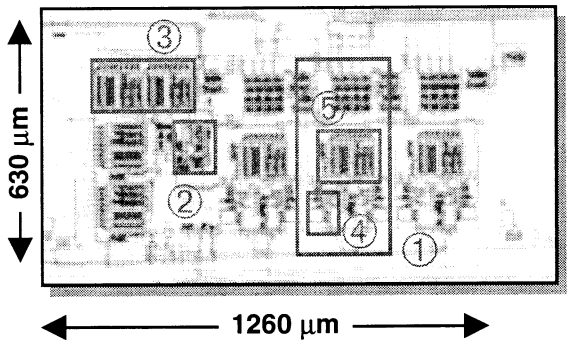


Fig. 4. MLP mixed-signal sequential test processor.

The FKBC controller features about 0.85 mm^2 (block 2 in Fig. 3) and is shown in more detail in Fig. 5. The main composing elements are: Four membership functions (1), three 2-input minimum circuits (2) [19], eight 6-bit mixed digital-analog multipliers (3), two analog translinear multipliers (4), two accumulating ATCs (5) and a pulse-width modulator (PWM) [20] used as a divider and output circuit (6), since the processor is targeted to control applications

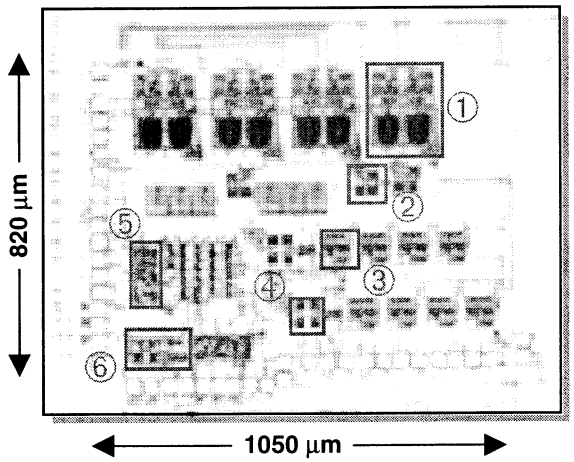


Fig. 5. FKBC mixed-signal sequential test processor.

4. COST-PERFORMANCE TRADEOFF

In this section we explore the design space for the MLP circuit as a design example that can be adapted to the application requirements of cost and performance. As a cost parameter we use an area occupancy index, whereas speed is used as the performance index, measured as the latency in clock periods needed by the processor to generate the output vector from an input vector.

Table III displays the measured area of the three key elements of the MLP processor. Let us assume a real application requirement of three layers of neurons, each layer including 10 neurons, and 10 synapses per neuron. This represents for Table I, $n_n = n_s = 10$ and $n_l = 3$. Using the expressions of that table with these figures, and calculating the area from Table III, we can plot each table row in an area-delay plot (Fig. 6). Of course, the number of possible combinations is much larger, however an interesting set of designs is already obtained.

8-bit A-D multiplier	sigmoid	ATC	
0.025	0.009	0.0135	mm^2

Table III. MLP key elements area.

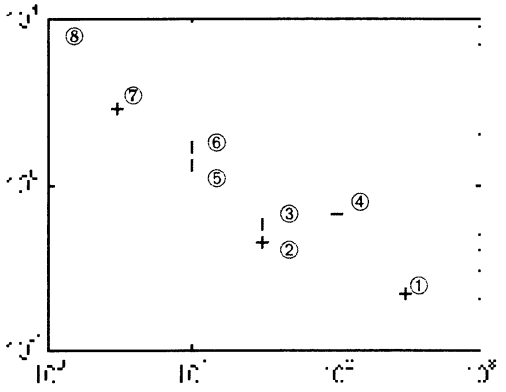


Fig. 6. MLP area-delay plot.

Although the plotted area figures are not directly the processor area, because glue logic and routing area is not accounted, the global area should be, with some deviation, proportional with the key element area. It is very interesting to see that, although some solutions are sub-optimal, there is a set of designs that trade off cost for performance. A hyperbolic curve of fairly constant area-delay product can be deduced as a straight line in the log-log plot of Fi.g 6. Hence, the most suitable option can be selected for a given application constraints.

5. CONCLUSIONS

Issues on sequential design of neural and fuzzy processors have been analyzed. Proper selection of sequentiality allows an application-specific efficient selection of the point in cost-performance (speed and area) design space. Furthermore, mixed-signal sequential processing exhibits high flexibility to emulate different system structures. Two sequential processors that show the feasibility of the approach have been designed and validated.

An important point is that sequentialization considerations are not technology independent, but take into account technology and subcircuit constraints. A consistent design style has been done.

The processors have been manufactured, and are currently being fully characterized. A quantitative experimental comparison in terms of speed, area and power consumption between the several solutions is being performed.

Acknowledgements. Work supported by CICYT Project No. TIC96-1195. Jordi Cosp holds a CIRIT Research fellowship.

REFERENCES

[1] Glessner M., Pöchmüller W., Neurocomputers, Chapman and Hall, 1994.
[2] Del Corso D., "Hardware Implementations of Artificial Neural Networks", Lecture Notes in Computer Science. J. Mira,F. Sandoval(Eds.), pp. 405-419, Springer-Verlag, 1995.
[3] Madrenas J., et al. "Analog and Mixed-Signal Neural VLSI Processors: Taxonomy, Comparison and Performance Evaluation", Proc. MIXDES'96, Lodz (Poland), 1996, pp. 389-400.
[4] Schalkoff R.J., Artificial Neural Networks, McGraw-Hill, 1997..
[5] C. Toumazou, F.J. Lidgey and D. Haigh, Editors "Analogue IC design: The current-mode approach", IEE Peter Peregrinus, London, 1991.
[6] Yazdi N. et al., "Pipelined Analog Multi-Layer Feedforward Neural Networks", Proc. ISCAS'93, pp. 2768-2771.
[7] Lee JC et al. "A Mixed-Signal VLSI Neuroprocessor for Image Restoration", IEEE Trans. on CAS, Vol 2. No.3. Sept. 29, pp. 319-324.
[8] Moreno J.M et al., "An Analog Systolic Neural Processor Architecture", IEEE MICRO, Vol.14, No.3, Jun. 94, pp.51-59.
[9] Moreno J.M. et al. "Analog Sequential Architecture for Neuro-Fuzzy Models VLSI Implementation", Proc. ICANN'97, Lausanne, Oct 1997, pp.1199-1204.
[10] Madrenas J., et al., "VLSI Design of a Flexible-Structure Sequential Mixed-Signal Neural Processor", MIXDES'99, pp. 259-264, Krakow (Poland).
[11] Moreno J.M. et al., "Systolic Modular VLSI Architecture for Multi-Model Neural Network Implementation", Proc. of the 4th Int. Conf. on Microelectronics for Neural Networks and Fuzzy Syst. pp. 118-124, 1994.
[12] Takagi T., Sugeno M., "Fuzzy Identification of Systems and its Applications to Modelling and Control". IEEE Trans. on Systems, Man and Cybernetics, Vol 15, n°1. Jan. 1985.
[13] E. Alarcón et al., "Implementation of an Application-Specific Fuzzy Controller by Means of a Mixed-Signal Sequential Architecture", Proc. IEEE MWSCAS'98, Notre-Dame, Indiana, Aug. 98.
[14] Toumazou C., et al., Eds. "Switched-Currents: An analogue technique for digital technology", IEE Peter Peregrinus, London, 1994.
[15] Espejo S. et al., "Tunable feedthrough cancellation in switched-current circuits", Electron. Lett., Vol. 30, No. 23, pp. 1912-1914, Nov. 94.
[16] Paulino N. and Franca J.E., "A CMOS digitally programmable current multiplier", Proc. IEEE ISCAS'96, May 1996.
[17] K.L. Fong and A.T. Salama, "A 10 bit semi-algorithmic current mode DAC", Proc. IEEE ISCAS'93, pp. 978.
[18] Bastiaansen C.A.A, et al.,"A 10-b 40-MHz 0.8μm CMOS current-output D/A converter", IEEE Jour. of Solid-State Circ., Vol. 26, N°7, July 91, pp. 917.
[19] Ramírez-Angulo J., et al., "Current-Mode and Voltage-Mode VLSI Fuzzy Processor architecture", Proc. IEEE ISCAS'95, pp. 1156.
[20] Alarcón E., et al., "Novel Pulse-Width-Modulated Current-Mode Analog Defuzzifier for the Fuzzy Control of Switching DC-DC Converters", 42nd MWSCAS. NMSU, Aug. 1999.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Simulations of construction learning as for neuron-computer resources

Tomoo AOYAMA , Hanxi ZHU, Tomoki TESHIMA, and Ikuo YOSHIHARA
The Faculty of Engineering, Miyazaki University
Gakuen kibanadai-nishi 1-1, Miyazaki 889-2192, Japan
e-mail: t0b217u@cc.miyazaki-u.ac.jp

Keywords: learning, neuron-computer, and image-shifter

Abstract:

Anyone would observe that information processing in the brain is depended on neuron systems. Simple neuron systems are neural networks, and their learning methods have been studied.
However, we concern about that investigations for large-scale neural network system are incomplete yet. Therefore, we study a scheme that primitive tiny networks are gathered and connected each other, then one functional logic are generated. We are sure that there is a potential to design a non Neumann-machine in the scheme. It is well worth consideration. In this paper, we discuss a construction method of large-scale neural-network systems.

1. Introduction

Information processing in the brain is complex and advanced extremely, so that we have believed. We are sure that there are many complicate structures behind such processing abilities. There are local structures called as "association cortex" in the brain, which have been known by medical and physiological investigations. But it is hard to discover any difference among the structures [1]. The detected structures are only common connections among the neurons; even if the functions of the structures are difference each other. We imagine that many kinds of the connections make such differences of functions in the local structures. If the hypothesis is true, why do the connections generate?

Anyone observes that information processing in animal brains is depended on neuron systems. A simplified neuron system is neural networks. Many kinds of the networks are known [2], whose functions are depended on connection structures among the neurons. The proper function of the neurons is primitive and it is easily modeled into an engineering neuron [3]. We must select one model of neural networks to derive discussions. The model is called as multi-layer neural network that has learning facility [4]. The facility is well worth consideration that we believe in. The network is a forward propagator, but by using the facility, any binary logic function can be realized. Any binary logic functions are represented as disjunctive normal forms [5]. The maximum product terms in the forms are constructed of logical variables and their not-operations. Therefore, if their "not-, product-, and junction-" operations are constructed of the networks, all binary logic units are realized by the learning. That is; accumulator, shifter, and many other computer resources can be generated as network-forms. Moreover, any true-value table is also expressed by the network. Therefore, the network represents many multi-value logic operators that are never expressed by the binary logic [6]. We are sure that there is a potential to design a non Neumann-machine on use of the neural network.

2. General logic in neural network

There is a neural network model constructed of multi-layer neurons [4], which is called as multi-layer or perceptron type neural network, and which has a function that translates vectors into different ones. Numbers of elements between the former and latter vectors are different. The translation is plural former vectors to one latter vector correspondingly, and is never one former to plural latter. The former vectors are called as input vectors or data, and the latter are teaching vectors or data. Their genetic name is learning data. Behavior of the multi-layer neural network is forward propagation orderly for the information represented by vectors. The property is important, for example, any binary logic functions (F) are represented as conjunctive normal forms: $F=X_1\&X_2\&X_3\&.....\&Xn$, where Xi (i=1,2,..) is maximum sum term, and "&" is the AND-operator. The sum term is constructed of logical variables and their not-operations. Therefore, following a network (Figure 1) can simulate any binary logics.

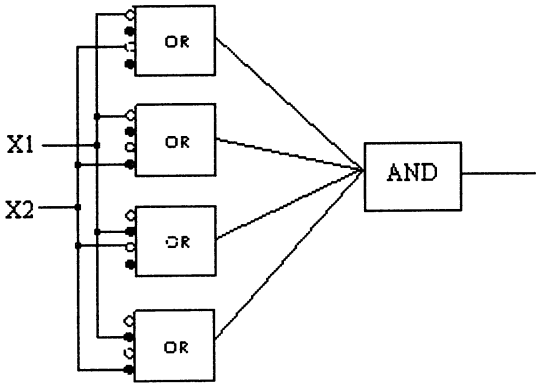


Figure 1. *Neural network representation for the maximum sum term:*

The figure is written by operation-pairs "OR,

AND", but a similar figure can be done by "AND, OR" also. In the case, we call it as "disjunctive normal form."

Where "$_{o}$ " and " • " are NOT and NULL operators. NOT, OR, and AND functions be simulated by neural networks. A differential form $\partial \mathbf{O}/\partial \mathbf{X}$ for the neural networks is known [7]. Where **O** is a vector for the output data, and **X** is for the input data.

$$\partial \mathbf{O}/\partial \mathbf{X}=\mathbf{V}\mathbf{f}'\mathbf{W}\mathbf{g}'$$

Where **V** is connection weights (a matrix) between first and second layers. **f'** is a vector representation of the differentials for input/output of the neurons on the second layer. **W** is a connection weights between second and third layers. And **g'** is a differential vector on the third layer. The multi-layer neural network is a kind of function, which has plural input and output. The function would replace a neuron that was also a function, i.e. constructs nest a set of neural networks. And the nest system can be learned by using $\partial \mathbf{O}/\partial \mathbf{X}$. The binding coefficients between the neural networks are calculated by iteration similar to the back-propagation algorithm.

4. Image shifter

4.1 Recurrent learning

The back-propagation is a useful algorithm, but it is applicable to rather tiny neuron systems. It is difficult to huge systems; moreover it sometimes doesn't converge on. We desired an algorithm to converge at any huge system, therefore introduced following recurrence of learning steps. We call the scheme as recurrence learning.

(1) The basic learning step is the back-propagation. A primitive logic unit is generated by the step. Interfaces of the primitive neural network have input, output, and special inputs/outputs for the next recurrence steps. The special inputs/outputs are "carry" signals in case of that the primitives are arithmetic resources. An example of true-value logic table for 2-bits digital shifter is following.

Table 1. 2-bits shifter

input			output		
c_in	d_in0	d_in1	out0	out1	c_out
A	B	C	A	B	C

Where A, B, and C are genetic letters for 0 or 1. c_in, d_in0, out0, and c_out mean carry-in, 0^{th} input data, 0^{th} output data, and carry-out, respectively.

(2) On each recurrence, interfaces of generated systems must have same attributions.

(3) On each recurrence, we fixed the number of interacting elements was 2. The interactions are connections among the carry-signals. When elements are 2, numbers of the selection are 2, where the learning is no longer iterative calculations, but only path-selections. And the path is fixed at any case, and then the recurrence learning [8] is completed. Even if the operation were path-selections, teaching data are necessary. A guess for initial path was determined randomly. The initial guess dependency is negligible. Next, we show an output list of the learning.

Recurrent learning output list for 256-bits shifter:

Italic part s are comments.

```
primary neural network, 11 neurons
tmpc=0.500000  tout[0]=0.500000  sf2l,le_sf2=1
can't learning, next trial
                    inappropriate path connection
connection completed, go to genaration
<sf2> layer=6, 22 neurons
<<4th shifter>>
                    This is processing for 4-bits shifter.
... layer=12, 44 neurons
<<8th shifter>>
... layer=24, 88 neurons
<<16th shifter>>
... layer=48, 176 neurons
<<32th shifter>>
... layer=96, 352 neurons
<<64th shifter>>
... layer=192, 704 neurons
<<128th shifter>>
... layer=384, 1408 neurons
                    Now, 256-bits shifter is constructed.
```

Using the recurrence learning, we can design logic-units scaled as 2**n, for example, 1024 bits shifter, 64 bits adder, and so on. We tried to generate 256-bits digital-shifter by using back-propagation only, but couldn't do. There are too many problems, i.e. learning data set, convergence of iterations, and CPU time. If the recurrent learning is adopted, the shifter will be generated within 1 second on use of Pentium-Pro 200 MHz.

In the BP learning, a primitive shifter was generated first, which was constructed by 4-, 4-, and 3-neurons. The neurons were on 1^{st}, 2^{nd}, and 3^{rd} layers, respectively. The number of neurons was set to be minimum. The neural network can shift digital signals of 2 bits. We call it as 2-bits digital-shifter. Next, we generated a 4-bits digital-shifter by using twice 2-bits digital-shifters. At the time, we used the recurrence learning. Then, we repeated similar schemes by 8 times, and got a 256-bits digital-shifter. The CPU time of the repeat schemes were almost negligible (~0.05 sec.) as compared with BP learning for a primitive shifter (1.0 sec.). A 256-bits shifter had 3*{2**(8-1)}=384 layers and 11*{2**(8-1)}=1,408 neurons.

4.2 Analogue shifter

Output signals of the primitive shifter are analogue. It is difficult that the signals are transferred into next primitives. Small errors on the signals are increased and finally actions for long-bits shifters are destroyed. We put a digitizer at output side of the neural networks after some trials. The digitizer was necessary, and it could be constructed by a neuron

having large-exponent sigmoid function without learning. The digitizer is also a neural network.

4.3 AD converter

Neural networks with the digitizer can't process analogue signals. We converted analogue signals to 8-bits digital ones by using AD converter. The BP learning for 8-bits AD converter was very difficult. The learning doesn't keep synchronization among output signals; therefore, it can't suppress ripples near crossovers. For suppression for the ripples, we divided the AD converter into two units. One unit converts analogue signals into step-values. Another unit does the step-values to 8-bits digital signals. The former unit is nearly equal to digitizer, and doesn't need learning. The later can be learned by BP. The AD converter is a neural network. Without these digitizer and AD units, the number of neurons for a 256-bits analogue shifter is 1,408*8=11,264.

4.4 Image shifter

An image shifter is constructed of plural n-bits shifter. So, why are the plural shifter resources generated? We have searched a learning method in order to generate lots of operation-resource, but can't discover it. Such a learning method might be nothing. Now, we assume that plural same functional elements can be generated at same time through the recurrent learning. This is a complete supposition, which many primitive resources are generated in first learning step, then, they are vanished away in rather short timing period. However, if a part of them were used in a second learning step (i.e. recurrent learning), it is held and fixed in a functional logic unit. There is no reference connected with the assumption.

Under the assumption, we constructed of an image shifter, which moved a full color image to xy-2-directions by any bits. The image shifter has two units that are same function, but their shifting directions are orthogonal. Learning method to make use of one unit as plural different uses was not found. However, we are sure that God doesn't senseless actions, and then we assume the existence of the method. We generated the one unit by using recurrent learning. The one unit includes 11,264*256*3= 8,650,750 neurons. Such huge neural networks would be never generated only by BP learning.

For the image-shifter (256*256 pixels), a response is listed in figure 2a and 2b.



Figure 2a. *An original image (256*256 pixels): This is a full color image; that was filmed at an ancient palace in Seoul, Korea.*



Figure 2b. *An image shifted by two neural networks, which have 8.6 million neurons respectively. The image is one after 7 pixels moved downwards and 13 pixels moved left-hand side. Black zones on upper and left sides show the result of the operations. CPU time was 5 min. CPU is Pentium-2 (300MHz), and OS/compiler are Linux/gcc. The details of original images are kept through 7*13*384=34,944th neuron layers.*

5. Conclusion

A function of multi layer neural networks is a converter between two vectors. That means any true-value table can be represented. We have a possibility making any information-processing unit by using the neural networks. However, it is hard to complete them practically. Back propagation learning is insufficient. It is applicable for small primitive units, but can't apply units including many neurons. We designed a learning method in order to construct huge neuron units. The method requires nesting logic structure. But, even if there is the restriction, the method is effective. We designed several kinds of computer-resources indeed, in which there was a full color image shifter for 256*256 pixels whose number of neurons is 8.6 million. The huge neural network can shift images to two directions by any bits, where details of the original are kept. We showed one sample in the binary-logic. We believe that huge scale neural networks have faraway possibility, as an example as following.

Neural networks can express any multi-value logic functions, which are not done by the binary

sufficiently. We believe that neural network representations of conjunctive or disjunctive normal form for the multi-value logic are well worth consideration. It may give us a key to design non-Neuman architecture machines.

We showed that complex functions were constructed of primitive neurons. However, you would observe a lack in the complex, which was a memory function [9]. We believe in the memory system in neural networks should be studied from various directions.

References:

[1] Eiichi IWAI, "The Brain, mechanisms of learning and memory in monkey and man (in Japanese)", Asakura Co. Ltd., Tokyo, 1984.11, ISBN4-254-10031-0. The book is a text for medical doctors and psychologists.

[2] K.Nakano, et al., "An Introduction to Neuro-computing (in Japanese)", Corona Publishing Co. Ltd., Tokyo, 1990.4, ISBN4-339-02276-4.

[3] The engineering neuron is represented by two factors, which are a function and a threshold. The function is $1/\{1+\exp(-\alpha x)\}$; that is a sigmoid function in general. But other continuous functions are also used, if they are differential. We are sure that selection of the function is an important factor in order to define neuron systems. The parameter α and the threshold are determined in the learning process.

[4] D.E.Rumelhart, J.L.McCleland and the PDP Research Group, "Parallel distributed processing", MIT Press (1986).

[5] Masao Mukaidono, "Fuzzy logic (in Japanese)", Daily industry news Co. Ltd., Tokyo, 1991.11, ISBN4-526-03290-5.

[6] Many multi-value logic operators are introduced in reference [5]. Some operators are never represented by the binary operators. That is their logic expressions can't be any binary one.

[7] T.Aoyama and H.Ichikawa, "Back propagation algorithm restricted with some conditions (in Japanese)", IPSJ Sig. Notes 91-NA-37 (1991.7). Hereafter, IPSJ: Information Processing Society of Japan.

[8] T.Aoyama, "Large scale multi layer neural networks", IPSJ Sig. Notes 99-HPC-76 (1999.5).

[9] H. Zhu, T. Aoyama, and I.Yoshihara, "Functional memories constructed of neural network", Proceedings of the 14th KACC, E-210 (1999.10.14).

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Ultra-Precision Position Control of a Piezoelectric Actuator Using Neuro-PD Controller

Byung-Ryong Lee, Se-Woong Jeong, and Kyu-Youl Park

School of Mechanical and Automotive Engineering
University of Ulsan
P.O.Box 18 Ulsan 680-749, Korea
E-Mail: brlee@uou.ulsan.ac.kr

ABSTRACT

A piezoelectric actuator is widely used in precision positioning applications due to its excellent positioning resolution. However, the piezoelectric actuator lacks in repeatability because of its inherently high hysteresis characteristic between voltage and displacement.

In this paper, a controller is proposed to compensate the hysteresis nonlinearity. The controller is composed of a PD and a neural network part in parallel manner. The output of the PD controller is used to teach the neural network controller by the unsupervised learning method. In addition, the PD controller stabilizes the piezoelectric actuator in the begining of the learning process, when the neural network controller is not learned. However, after the learning process the piezoelectric actuator is mainly controlled by the neural netwok controller. In this paper, good tracking performance of the proposed controller was verified by experiments and was compared with the classical PD controller.

Keywords: Piezoactuator, Neural Net, PD Control, Hysteresis, Precision Control

1. INTRODUCTION

Precision positioning systems, such as scanning tunneling microscopy and diamond turning machines, is often required that actuators have nanometer resolution in displacement, high stiffness, and fast frequency response. In general, piezoactuators are used to meet the above requirements. However, piezoactuators have some limitation that they are in lack of accuracy due to hysteresis nonlinearity and drift. It is reported that the maximun error due to hysteresis nonlinearity can be as much as 10 to 15 percent of the desired path if the actuators are run in open loop manner. Because piezoceramic materials are ferroelectric, they fundamentally exhibit hysteresis behaviour in their response response to an applied electric field. There have been much researches in order to improve the hysteresis behaviour of piezoactuators.

Jung and Kim[1] considered that the nonlinearity problem of piezoactuators are mainly based on the electrical nonlinearity and the hysteresis. They tried to suppress the electrical nonlinearity by adjusting the electrical condition based on hardware device, then they introduced mathematical modeling of hysteresis behaviour. Newcomb and Flinn[2] proposed that charge-based control input instead of voltage input can reduce the hysteresis nonlinearity of the piezoactuators. The control system in the approach can be simple because it is possible to use the linear control theory. However, the hardware requirements are increasing and a special charge-drive-amplifier should be designed, and also the sensitivity of displacement was decreased. Kaizuka and Sui[3] proposed a hysteresis reduction scheme, in which piezoactuator and capacitor are connected in serial to control the system in linear fashion. However, in this case much higher voltage input must be supplied to produce the displacement output. Another approach to reduce the hysteresis behaviour is to use feedback control algorithm[4], where the hysteresis behaviour is not included in the control system. Therefore, the control stability may be unstable and control accuracy can not be guaranteed. Ge and Jouaneh[5] introduced the classical Preisach algorithm to model the hysteresis behaviour of the actuators, but in his scheme the tracking performance is guaranteed for the periodic trajectory. The proposed schemes above are usually to apply the classical control scheme after designing the mathematical modeling of the hysteresis nonlinearity or analysing the dynamic equation of the piezoactuator system.

In this paper we proposed the neuro-PD parallel controller[6][7] in order to reduce the hysteresis nonlinearity of the piezoactuator. The main advantage of the scheme is to abbreviate both modeling of the hysteresis and analysing the dynamic system of the actuating system, instead the dynamic behaviour as well as hysteresis nonlinearity can be self-learned in real time. An experimental equipment was established to show the

feasibility of the proposed control scheme, and the tracking performance of the system was compared to that of simple PD controller.

2. NEURO-PD PARALLEL CONTROLLER

In this paper, so called neuro-PD parallel controller which consists of multi-layer neural network and classical PD controller in parallel manner. Fig.1 show the schematic diagram of the controller.

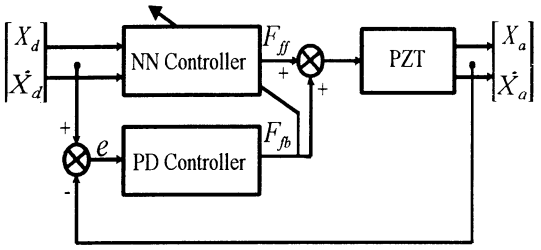


Fig. 1 Structure of controller

In Fig.1, the control inputs of the neuro controller are the displacement and velocity of the piezoactuator system. The inputs to the PD controller are the displacement and velocity errors, and the output of the PD controller is the feedback force F_{fb}. The neuro controller is self-learned by the displacement and velocity of the piezoactuator system. The objective function for learning process is simply the 2 power of F_{fb} value. In learning process, the weights of each layer of the neural network are adjusted so as to minimize the objective function. The output force F_{fb} of the PD controller is used to train the neural network via learning process. The objective function $E(t)$ that means training error can be expressed as following equation:

$$E(t) = \frac{1}{2} \| o_k(t) - y_k(t) \|^2 \quad (1)$$

,where $y_k(t)$ and $o_k(t)$ are the desired and actual output value of k^{th}output layer in t^{th} step, respectively. In general, when the desired output values of the output layer are known, it is possible to train neural network by supervisory learning method. However, in this case the value $y_k(t)$ can not be calculated directly, so neural network should be trained in unsupervisory learning method. The objective function of Eq.1 can be substituted as following equation:

$$E(t) = \frac{1}{2} \| F_{fb}(t) \|^2 \quad (2)$$

,where $F_{fb}(t)$ can be defined as

$$F_{fb}(t) = K_p(X_d - X_a) + K_d(\dot{X}_d - \dot{X}_a) \quad (3)$$

In Eq.3 X_d and $\dot{X}_d$ are the desired displacement and velocity of piezoactuator, respectively, and $X_a$and $\dot{X}_a$ are the actual displacement and velocity of piezoactuator, respectively. In fact, Eq.1 and Eq.2 has the same meaning. In other word, the fact that the training error approach to zero means that displacement error of the system also approach to zero. Therefore, it can be seen easily that if the displacement and velocity errors approach to zero, $F_{fb}(t)$ value naturally approach to zero. In this paper, backpropagation learning algorithm was used to train the neural network. During the learning process, the objective function E(t) is minimized to a local minimum.

3. EXPERIMENTAL SETUP

Experimental setup for ultra-micropositioning consists of the piezoactuator of Physik Instrumente, Capacitive Sensor, and data acquisition board. The detail specification of the system is shown in Table.1, and the schematic diagram of the experimental setup is shown in Fig.2. In the figure, desired position data is delivered to the D/A converter, and the analog voltage is applied to the amplifier of the piezoactuator, where the amplified voltage between 0 to 1000 V is applied to piezoactuator.

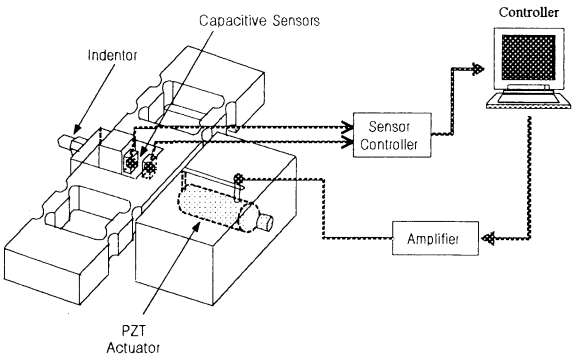


Fig. 2 Schematic Diagram of PZT system

Table. 1 Main specification of experimental apparatus

	Model	Remark
PZT	PI P-245.20	Max.stroke: 0 ㎛
PZT Amp	PI E-471.00	0 - 1000 V
Capacitive Sensor	PI D-050.00	25㎛ ~ 75㎛
Capacitive Sensor Controller	PI E-509.C1	1 channel

4. EXPERIMENT AND RESULTS

In the experiment, first in order to find the the hysteresis characteristic of the piezoactuator, input voltage applied to the piezoactuator was increased from 0 to 1000 V , next the input voltage was decreased from 1000 to 0 V in the open loop control. Fig.3 shows the hysteresis result of the piezoactuator used in the experiment. In this paper, a neuro-PD parallel controller, as shown in Fig.1, was introduced to compensate the nonlinearity effect due to the hysteresis problem. The neuro control part of the controller is trained, with the output value of the PD control part of the controller, to learn the dynamic characteristics of the piezoactuator system. In this paper, a sinusoidal function with 3 second period and 3 *um* amplitude was chosen as the desired trajectory to train the neuro control part of the controller.

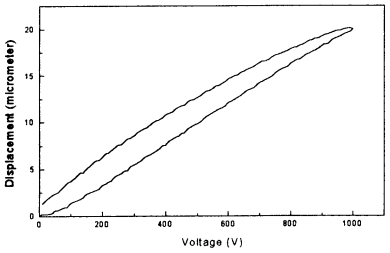
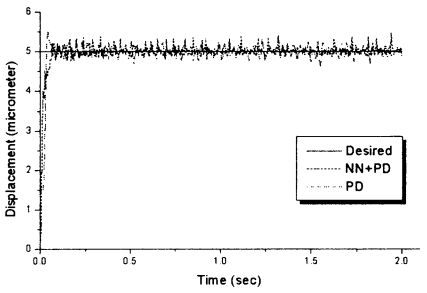
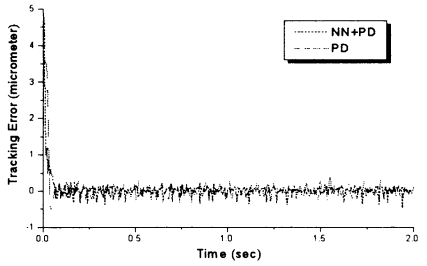


Fig. 3 Hysteresis of PZT

The neural network consists of an input layer with 2 nodes, an hidden layer with 8 nodes and an output layer with 1 node. The sampling time to learn the neural network is 0.005 second, which means the desired sinusoidal trajectory is partitioned into 600 pieces in one learning cycle. During the learning process the weights in the neural network are changed to minimize the training error which is defined in Eq.2. In our case, the learning error rate was reduced to 6 percent after 200 times of learning cycles. After learning process was finished, an experiment was carried out to test the trajectory tracking performance of the neuro-PD parallel controller, and the performance was compared with the case of PD only controller. In both case, the used gains for PD controller are Kp=0.2, and Kd=0.000005. First, a step response was compared. As shown in Fig.4, PD only controller has faster response than Neuro-PD controller, but the former gives an overshoot in the beginning. Next, a trajectory following performance was compared. In this case, the given trajectory is y=5+3*sin(2/3*PI*t). As shown in Fig.5 and Fig.6, the tracking performance of neuro-PD controller is better than that of PD only controller. In experiment, the control output values of neuro and PD control part of the neuro-PD parallel controller are compared. It is shown that the neuro control part of the neuro-PD controller have a major effect to control the piezoactuator system. However, neuro control part gives much error when the dynamics of the piezoactuator system is changed due to external noise or disturbance, in that case the control output of PD control part can be increased to compensate the error.

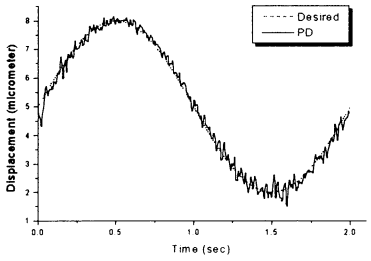


a) Displacement

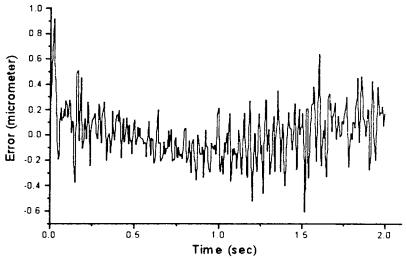


b) Tracking Error

Fig. 4 Tracking Performances of PD and PD + NN for a Step Input

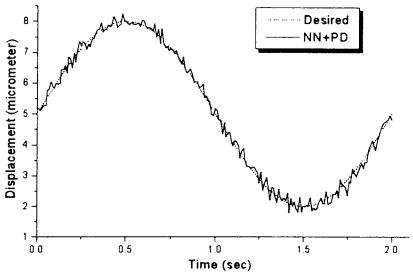


a) Displacement

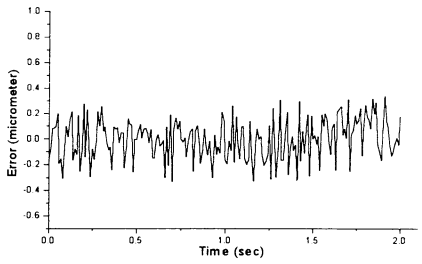


b) Tracking Error

Fig. 5 Tracking Performance of PD controller
for Sinewave Trajectory



a) Displacement



b) Tracking Error

Fig. 6 Traking Performance of NN+PD controller
for a Sinewave Trajectory

5. CONCLUSIONS

In this paper a neuro-PD parallel controller, which exploits the advantages of neuro and PD controller, was applied to control the piezoactuator system which inherently has hysteresis nonlinearity. The main advantage of the controller is that it is not necessary to model the dynamic equation or hysteresis nonlinearity of the piezoactustor system. In fact, the piezoactuator dynamics and hysteresis model are learned in self-learning fashion in the neural network. In order to show the performance of the neuro-PD controller, several experiments were performed. In conclusion, it is seen that neuro-PD controller gives better performance than PD only controller. In this paper, the average tracking error of the piezoactuator system is about 50 nm. In future study, a better intelligent control algorithm and hardware performance of the measuring sensor will be applied to control system, which will finally improve the tracking performance of the system.

ACKNOWLEDGMENT

This research was supported by Research Center for Machine Parts and Materials Processing(REMM) at University of Ulsan.

REFERENCES

[1] S. Jung and S. Kim, "Inprovement of scanning accuracy of PZT piezoelectric actuators by feedforward model-reference control",Precision Eng. P. Ge and M. Jouaneh, Vol.16, No.1, pp.49~55, 1994

[2] C. V. Newcomb and I. Flinn, "Improving the Linearity of Piezoelectric Ceramic Actuators", Electroncs Letters, Vol.18, No.11, May, pp.442~444. 1982

[3] H. Kaizuka and B. Sui, "A simple way to reduce hysteresis and creep when using piezoelectric actuators", Japan J. Appl. Phys, Vol.27, No.5, pp.773~776, 1988

[4] M. Tanaka, Z. W. Jiang and S. Chonan, "Force Control of a Flexible Finger with Distributed Force Sensors and Piezoelectric Actuators", Proceedings of the 1st International Workshop on Advanced Mechatronics, pp.237~241, 1995.

[5] P. Ge and M. Jouaneh, "Tracking Control of a Piezoceramic Actuator", IEEE transactions on control systems technology, Vol. 4, No. 3, 1996.

[6] Q. H. Max Meng, "A Neural Network Adaptive Controller For Robots with Unknown Dynamics", Proceeding of 1993 International Joint Conference on Neural Networks.

[7] T.Yabuta, and T. Yamada, "Possibility of Neural Networks Controller for Robot Manipulator", IEEE, 1990.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Parameter Identification and Trajectory Control of a DC Motor Using Neural Networks

Dong-Seog Bae
Dept. of Automatic production
Chang-Won Polytechnic Collage
Chang-Won 641-777 Korea
dongsbae@cw-polytec.ac.kr

Jong-Il Bae
Dept. of Electrical Engineering
Pukyung National University
Pusan 608-739 Korea
jlbae@elet.pknu.ac.kr

Jang-Myung Lee
Dept. of Electronics Engineering
Pusan National University
Pusan 609-735 Korea
jmlee@hyowon.pusan.ac.kr

Abstract

This paper introduces a high-performance speed controller based on artificial neural networks. The goal of this research is maintaining the speed of a DC motor to follow an arbitrary selected trajectory. Furthermore, it is pursued to obtain an accurate trajectory control of the motor, especially under the situation that the parameters of motor and load are unknown. Using an artificial neural network, the unknown nonlinear dynamics of the motor and the load can be estimated. To form a trajectory controller, the trained neural network identifier is combined to a reference model that represents the control characteristics of the motor. The performance of the identification and the control algorithm are evaluated by the simulation and experiment. A typical DC motor model and the trained nonlinear dynamics of the load are used for the simulation and experiment. The results show that this algorithm is robust against parameter variations of the load and the motor.

Keywords : unknown parameter, artificial neural network, DC servo motor driver

1. Instruction

In an electric driving system, one of the important problems of traditional tracking controller is that unknown load parameters which have wide operating point can be acquired. This coincides with the problems of each controllers[1-4]. There are many techniques to overcome such problems. Therefore identifying the whole nonlinear system through a linearized model around the widely varying operating point with a high speed switching frequency can introduce an error inducing unstable or inaccurate performance[4].It is suitable to apply neural network for training Non-linear function[5][6].

Neural networks are implemented properly set on input-output pattern which output by plant and trained for emulating dynamics of unknown non-linear plant. If system dynamics is identified by ANN, Traditional control techniques can apply for reaching aim that they wants. One of them, indirect model reference adaptive control (MRAC)[5][6], is useful to application of trajectory control especially. In this paper, we propose the method on control topology based on MRAC techniques for trajectory control and identification of DC. servo motor based on ANN. This paper consists of as followings. In section 2 dynamics of DC motor and necessary functions for ANN training are described. In section 3 simple introduction for ANN simulator and how function imitates are presented. In section 4 we measured the performance of identification of DC motor. In section 5 we introduced the controller topology about identified model.

2. DC motor model

1. Motor dynamics

Dynamics of DC motor is calculated by following two equations.

$$K\omega_p(t) = -R_a i_a(t) - L_a[di_a(t)/dt] + V_t(t) \quad (1)$$

$$Ki_a(t) = J[d\omega_p(t)/dt] + D\omega_p(t) + T_L(t) \quad (2)$$

Load torque, $T_L(t)$, can be expressed as function of rotor angular velocity as following.

$$T_L(t) = \Psi(\omega_p) \quad (3)$$

Function $\Psi(\omega_p)$is dependent of load and accurate function is assumed to be unknown.

2. Discrete time DC motor model

In order to acquire training data of ANN and apply control algorithm, discrete time DC motor model is nessarey.

$$T_L(t) = \mu\omega_p^2(t)[\mathrm{sign}(\omega_p(t))] \quad (4)$$

Discrete time model can be established by substituting all continuous differential as finite

difference after gathering equation (1),(2),(3),(4). State-space representation is as followings.

$$\omega_p(k+1) = \alpha\omega_p(k) + \beta\omega_p(k-1) + \gamma[\mathrm{sign}(\omega_p(k))]\omega_p^2(k) + \delta[\mathrm{sign}(\omega_p(k))]\omega_p^2(k-1) + \zeta V_t(k) \quad (5)$$

1HP, 220 V, 1100 RPM DC motor is used in simulation. And parameters are as follows.

$J = 0.068\ \mathrm{Kgm}^2$
$K = 3.475\ \mathrm{NmA}^{-1}$
$R_a = 7.56\ \Omega$
$L_a = 0.055\ \mathrm{H}$
$D = 0.03475\ \mathrm{Nms}$
$\mu = 0.0039\ \mathrm{Nms}^2$
$T = 40\ \mathrm{ms}$

3. ANN and Back propagation learning

An ANN is trained to emulate a fuction by presenting it with a repesentative set of input/output functional patterns.

The backpropagation learning method is minimized the difference between the desired output and the real output for all the given pattern to adjust the weight in the link and the threshold in the node. Because P^{th} training pattern $(P=1,.....P)$ minimize following the energy function for all the weight and threshold.

$$E_p = (1/2)\ \Sigma_i (T_i - Y_i(H))^2 \quad (8)$$

$Y_i(H)$ match the output layer in H with the action of i^{th} neuron. Also, T_i is desired value. it computed to match update value with the weight by gradient descent technique.

$$w_{ij}^{new}(h) = w_{ij}^{old}(h) + \eta\partial E_p/\partial w_{ij}(h) + \upsilon\triangle w_{ij}(h) \quad (9)$$

the quantity for $\partial E_p / \partial w_{ij}(h)$ is computed by following the presentation.

$$\partial E_p / \partial w_{ij}(h) = \delta_i(h) f[u_i(h)] \cdot Y_j(h-1) \quad (10)$$

$$1 \le i \le N_{h+1}$$

$$\delta_i(h) = \Sigma_j \delta_j(h+1) f[u_i(h+1)] \cdot w_{ji}(h+1) \quad (11)$$

where, $\delta_i(H) = -(T_i - Y_i(H))$

The enough learning is accomplished when the sum of the total error function that is included in the collection of all the P training pattern is decreased under the threshold ε that is selected beforehand. Namely,

$$E_{total} = \Sigma_p E_p < \varepsilon, \qquad p=1,....P \quad (12)$$

More details about backpropagation algorithm are commented in reference[5].

4. ANN for system identification and control

Fig. 1 shows the basic concept about identification of DC motor and control $\varepsilon_i(k)$ and $\varepsilon_c(k)$ of figure 2 are defined as discrimination and tracking error, respectively. The aim or discrimination is to minimize the error.

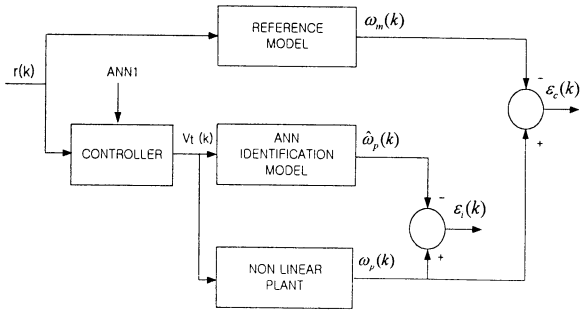


Fig. 1. ANN Based identification and control system.

$$[\varepsilon_i(k)]^2 \ \forall\ kT \in [0, t_f]$$

Or, we minimize $[\varepsilon_i(k)]^2$ in all.

Here control strategy is to calculate the $kT \in [0, t_f]$ most optimal terminal voltage $V_t(k)$, and this minimizes tracking error

1. Identification of DC motor

The charicteristics of DC motor are identified by making the set of input and output pattern for ANN. We constrain the operation space as follows to unite the limit of mechanical and electric hardware with imaginary scenario

$$-30.0 < \omega_p(k) < 30.0\ \mathrm{rad/s}$$

$$|\omega_p(k-1) - \omega_p(k)| < 1.0\ \mathrm{rad/s}$$

$$|V_t(k)| < 100\ \mathrm{v}$$

The performance of ANN discrimination is appraised by comparing the estimated output and real output of motor to normal arbitrary excitation signal.

2. Topology

Equation (5) becomes as equation (13).

$$V_t(k)=g[\omega_p(k+1),\ \omega_p(k),\ \omega_p(k-1)] \quad (13)$$

Here, $g[\cdot]$ is derived from equation (11) and we suppose that unknown.

$$\begin{aligned} g[\omega_p(k+1),\ \omega_p(k),\ \omega_p(k-1)]=&[(\omega_p(k+1)-\alpha\omega_p(k) \\ &-\beta\omega_p(k-1)-\gamma\,[\mathrm{sign}(\omega_p(k))]\omega p^2(k) \\ &-\delta[\mathrm{sign}(\omega_p(k))]\omega_p^2(k-1)/\xi \end{aligned} \quad (14)$$

ANN training to imitate $g[\cdot]$. $\omega_p(k+1)$, $\omega_p(k)$, $\omega_p(k-1)$ are independent variables of $g[\cdot]$ and inputs of ANN. The corresponding desired output $g[\omega_p(k+1), \omega_p(k), \omega_p(k-1)]$ is calculated from equation(14). This is also equal to $V_t(k)$ of equation(13). Desired output $V_t(k)$ corresponding to input pattern of randomly raising $[\omega_p(k+1), \omega_p(K), \omega_p(K-1)]$ is used for off-line training. Training data come from operation area defined in 4.1.

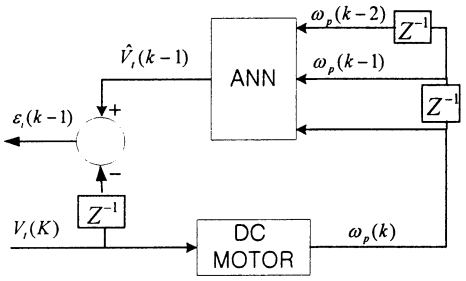


Fig. 2. Structure of the ANN identifier.

Fig. 2 shows model structure of serial/ parallel ANN discriminator.

$$\hat{V}_t(k-1)=N[\omega_p(k), \omega_p(k-1), \omega_p(k-2)] \quad (15)$$

The following statistic describes the simple concept including training result and ANN Topology.

Number of input : 3
Number of output : 1
Number of hidden layer : 1
Number of hidden neuran : 6
Number of training Patterns P : 1500
Number of training Sweep : 5000
Training Step η : 0.1
Momentum gain ν : 0.4
Total threshold ε of E : 0.01

The performance of trained ANN discriminator is estimated from exciting the DC motor model by using the following voltage sequence and compared with output of equation (15) and $V_t(k)$.

$$V_t(k)=50\sin(2\pi kT/6)+45\sin(2\pi kT/4)\ \forall\ kT\in[0,20] \quad (16)$$

Here, is restricted in $kT\in[0,20]$. Figure 6 shows the result.

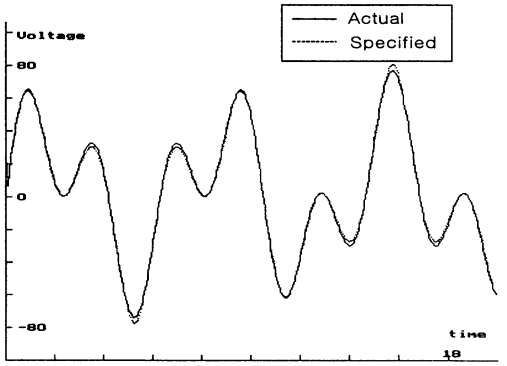


Fig 3. Actual and estimated terminal voltages of the DC motor

5. Experiment of DC motor using trained ANN

1. simulation

The purpose of this control system is that the velocity of motor tracks the desired trajectory $\omega_m(k)$ using DC motor. The following second-order reference model is chosen.

$$\omega_m(k+1)=0.6\omega_m(k)+0.2\omega_m(k-1)+r(k) \quad (17)$$

$r(k)$ is the bounded input to the reference model. The control sequence $r(k)$ can be calculated by using equation (17) corresponding to desired sequence $\omega_m(k)$.

In section 4 trained ANN is used at Topology to track input voltage $V_t(k)$ of DC motor, and guarantees that DC motor velocity, $\omega_p(k)$, correctly pursues trajectory.

Because selected reference model stabilize asymptotically, we assume that tracking error moves toward "0", and can predict velocity att $(K+1)^{th}$ sampling time in the equation (18).

$$\hat{\omega}_p k+1=0.6\omega_p(k)\ +\ 0.2\omega_p\ +\ r(k) \quad (18)$$

The result can be fed back to ANN that was learned in Topology in oder to estimate the control input at **K**th sampling time.

$$\hat{V}_t(k) = N[\hat{\omega}_p(k+1),\ \omega_p(k),\ \omega_p(k-1)] \quad (19)$$

Overall structure and control system for identification based on ANN is shown in Fig.4.

Referring Fig.5., while we acquired tracking performance on reference velocity trajectory with sinusoidal wave, we acquired and compared the real performance on velocity trajectory with sigmoid type in Fig.6. From Fig.5 and Fig.6 we can know that tracking performance is better. In addition, suggested algorithm has advantage that DC motor is independent to the load parameter. Here to get the improved performance, ANN is in need to be trained more.

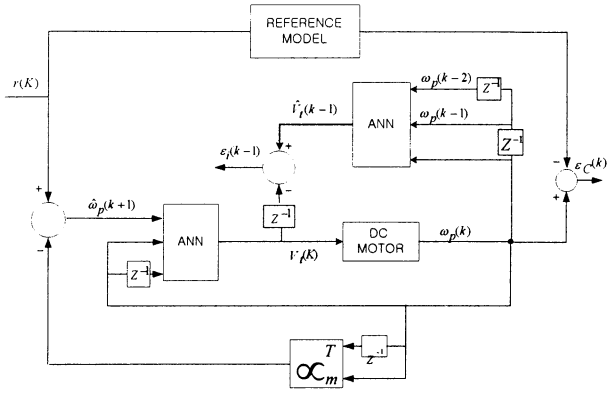


Fig. 4. Overall structure of the controller.

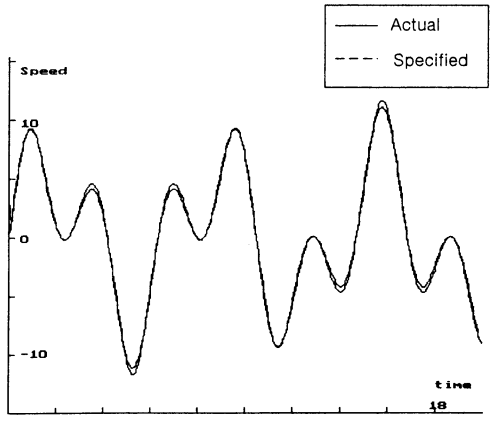


Fig. 5. Tracking performance for a sinusoidal reference trajectory.

Electrical driving system must show the superior characteristics in the noisy environment. In order to investigate the performance of the controller under the noisy environment, aftermixing uniform random variable of ±2 rad/s with the content in Fig.6, we repeatedly measured the velocity, $\omega_p(K)$ of DC motor. Here, we used the velocity $\omega_p(K)$

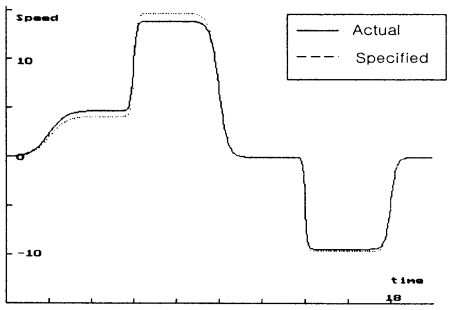


Fig. 6. Tracking performance for a sigmoid reference trajectory.

$\omega_p(K)$ as the input of ANN controller. Fig.7 shows the tracking performance in environment with noised continuously generated in Topology. Comparing Fig.6, tracking accuracy was continued under active mode including a noise and observed a stabilized controller performance.

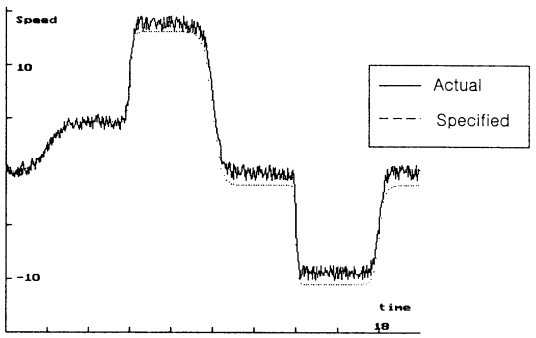


Fig. 7. Tracking performance for a sigmoid reference track with measurement noise.

2. The experiments in hardware

The proposed control algorithm composed of the controller which maked a program in the personal computer(PC). The control output signal is changed the analog voltage(0V-10V) by the 12bit D/A convertor(ACL-711B) in the PC. This signal is used to the forward/backward circuit of motor that is translated into the range of voltage, 5[V].

The velocity control of motor made use of the PWM method and element of electric power, IGBT. The analog signal generated the velocity control of motor supplies the PC the digital value in the A/D convertor. The specification of DC servo motor used in experiment is as following.

Max input voltage : $40V_{DC}$
Min input current : 3.2A
Max velocity : 1100 RPM
Torque : 11.5Kg · Cm
Taco generator output : 3V/1,000 RPM

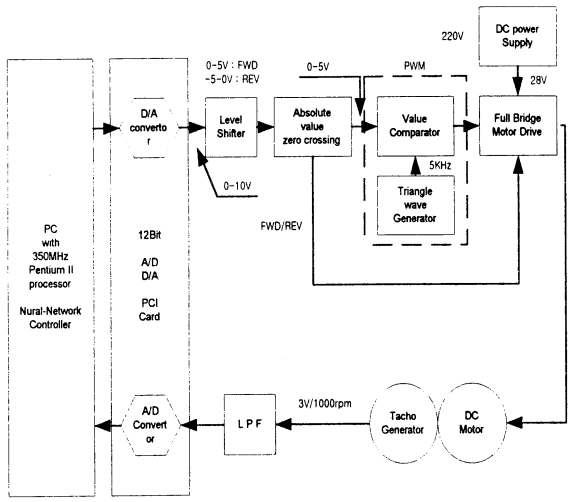


Fig. 8. Overall structure of experimented control system

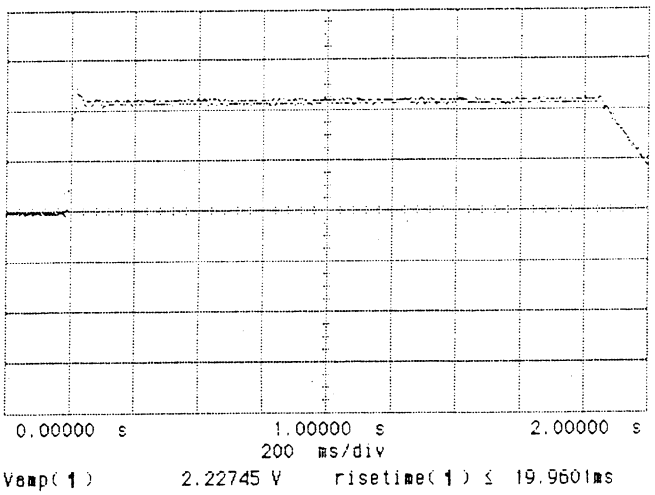


Fig. 9. Tracking performance of control algorithm to DC motor

The velocity of DC servo motor used to the rectangular wave of the 850 RPM and period of sampling, 1.64[ms] and made an experiment on disturbance that have a impulse about 10 Kg by periods. In steady state, the error can find to appear a deviation about 40 RPM.

6. Conclusion

It was suggested the controller using the ANN that executed the precise trajectory control of unknown DC servo motor. In the study, the nonlinear behavior unknown time invariant DC servo motor and the load showed that could be found by the ANN.

Also, it could be derived the advanced control property to make controller with the learning ANN that was applied the concept of model reference adaptive control for the trajectory control of rotor velocity. The tracking performance of the proposed control algorithm found robustness for the load or the variation of parameter in DC servo motor as a result of the simulation and the experiment in hardware. Also, To study the robustness of controller, it executed the experiment added to the variable noise. In the application, there would be considered necessarily the existence of noise. Specially, the experiment was found to have only the deviation about 40 RPM and to robust for the disturbance, 1Kg in hardware. It is important the assignment to apply this control algorithm to the application of robot manipulator, etc. at the further research.

Reference

[1] D. Miche, " Learning from human,s strategies of motor control: a challenge for robotic systems design," *5th IEEE International workshop on Robot and Human Communication,* pp. 223-228, Nov., 1996.

[2] K. S. Narendra and K. Parthasarathy, "Identification & control of dynamical systems using neural network." *IEEE Transaction on Neural Network*, vol. 1, pp. 4-27, Mar., 1990.

[3] S. Weerascooriya and M. A. El-Sharkawi, "Adaptive tracking control for high performance DC drives," *IEEE Transaction on Energy Conversion,* vol. 5, pp 122-128, Sept., 1989.

[4] S. Weerasooriya and M. A El-Shakawi."Identification and control of a DC motor using back-propagation neural network," *IEEE Transaction on Energy Conversion,* vol. 6, pp. 663-669, Dec., 1991.

[5] Jacke M. Zurada. Introduction to Artificial Neural Systems, pp. 256-358.

[6] Mohammad Bahrami Keith E. Tate, "Neural networks as complementary direct controller of nonlinear plants, Univ. of New South Wales. *Proc. of the 1993 IEEE International Symposium on Intelligent Control,* pp. 122-126, Aug., 25-27, 1993, Chicago, Illinois, USA.

[7] Cincotti S, Marchhesi M, Serri M, "A neural network model of parametric non-linear hysteretic inductors." *IEEE Transactions on Magnetics,* Vol 34, No. 5, P.1, pp. 3040-3053, Sept., 1998, Italy

[8] "Mietek A. Brdys, Grzegorz J. Kulawski, "Dynamic neural controllers for induction motor", *IEEE Transactions on neural networks,* Vol 10, No. 2, pp. 340-355, Mar., 1999

[9] Kshayar Pakdamam, Jean-Francios Vibert, Eric Boussard and Noureddine Azmy, "Single neuron with recurrent excitation: Effect of the transmission delay," Neural Networks, Vol. 9, No. 5, pp. 797-818.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Emergence of Cognitive Space in Rat Navigation —A Neural Network Model of the Shortcut Problem

Yoshito AOTA*, Yoshihiro MIYAKE+ and Seiji UKAI**

*+ *Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology*
4259 Nagatsuta-cho, Midori-ku, Yokohama-shi, Kanagawa-ken, 226-8502, Japan
** *Faculty of Engineering, Yokohama National University, Yokohama-shi, 240-8501 Japan*

Abstract

It is known that rat can take a shortcut even if it is an inexperience path for the rat. In this paper, we present a neural network model that selects a shorter path to an assigned goal in a given maze even if the model has only experience of a roundabout route. The model learns relationships of egocentric views on one or more environments by the recurrent network, which is equipped with the neurogenesis. When the model meets with a novel environment, these relationships are utilized for placing itself in the maze to search the goal through a shortcut.

1 Introduction

It is known that a rat is able to solve the navigation problem. For example, a rat can choose a shorter way in an unfamiliar environment, even if it is an inexperienced way for the rat (Tolman et al[1]). In case the information on the environment is incomplete for the rat, it is considered that the rat infers goal directions from utilization of the characters of the space that was acquired by the rat through experiencing various environments, i.e. from the cognitive map (Tolman[2]), and chooses a proper passage.

According to the reviews about biomimetic navigation models by Trullier et al[3], Trullier and Meyer[4], such performance on spatial tasks of rats is called the "metric navigation". Because the metric navigation needs some allocentric information on distances and angles between places, almost all models proposed so far use vector manipulations and coordinate frames that are implemented *a priori* in those models. In this paper, we define "coordinate system" (or "map") as some system by which the metric navigation (shortcut) is possible.

But, do real rats have 2D maps by nature? Although rats have their eyes opened and began exploratory behavior in 14 days after birth (Bolles and Woods[5]), it is estimated to be 21st day after which the spatial learning begins (Raunch and Raskin[6]).

On the other hand, recently the study of neurogenesis of adult animals has developed rapidly (Kempermann and Gage[7]). Neurogenesis means the generation of neurons in brains. In the case of rats, the hippocampus is known as the field where the neurogenesis happens (Altman and Das[8]). It would especially happened in a novel environment enriched with many attracting materials for rats (Kempermann, Kuhn and Gage[9]). Moreover, it is widely noted that the hippocampus forms the cognitive map (O'Keef and Nadel[10]).

So, we present a neural network model that forms a coordinate system through experiencing some environments by introducing the neurogenesis, which enables the model to predict a goal direction. The emergent process of the coordinate system is interesting for understanding the principle of information processing of the brain. Moreover, a wide application of such global optimality system is expected.

2 Model

2.1 Global structure of the model

As illustrated in Figure 1, the model is composed of 4 layers and one reward cell. The V-layer represents views from the model (visual information). The M-layer is constructed from motor neurons. Each motor neuron is encoding egocentric direction (i.e. a movement to the forward, the left etc). In addition to that, activities at the M-layer provide constraints for selecting recurrent connections between cells in the F-layer.

Since the F-layer has recurrent connections, the F-layer can have two modes. First is a learning mode, which constructs a coordinate system and establishes a relationship between the given environment and the coordinate system. Second is a reflection mode, which generates an adequate prediction of the goal through the RW cell and the W-layer.

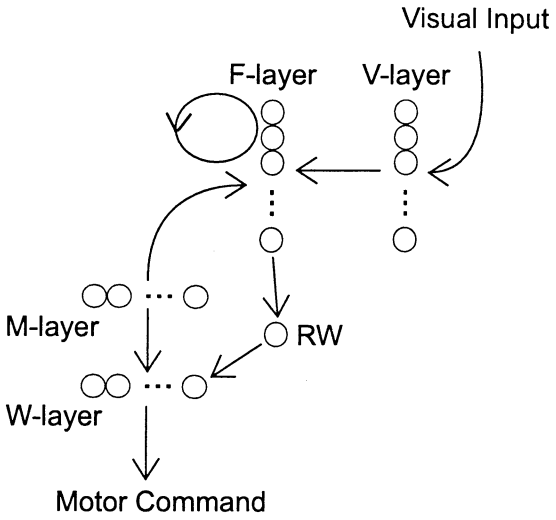


Fig.1. Structure of the model

2.2 Environments and views

Each "environment" consists of 3×3 measure as illustrated in Figure 2. "S" means the start position and "G" means the assigned goal. Each environment has a set of views. Each view *mv* is decided by an equation that is composed of the model's position (x,y) and allocentric direction *dr*. But coefficients are different from each other in environments (Fig.2). The *mv*-th cell of the V-layer becomes active.

2.3 Simulation protocol and formal description of the model

Step 0. Initial condition

Fix the total number N_F (=300) of neurons of the F-layer and N_V (=250) of the V-layer. The total number of neurons of the M-layer and the W-layer is 8, equal to the total directions. The output of the i-th neuron of the F-layer, Y_F^i, is 1 if $i=i_0=N_F$, and 0 otherwise. As initial values, connections from the neuron j of the V-layer to the neuron i of the F-layer, w_{FV}^{ij}, are random values between 0 and 0.1 ($\forall i,j$). Connections from the neuron j of the F-layer and the neuron k of the M-layer to the neuron i of the F-layer, w_{FFM}^{ijk}, are 0 ($\forall i,j,k$), and connections from the neuron i of the F-layer to the RW cell, w_{RF}^{i}, are 0 ($\forall i$).

Place the model into the start position, decide the model's allocentric direction, *dr*, and the egocentric direction, *m*.

Simulations consist of a number of trials. During each trial, the model carries out following steps.

Step 1. Learning mode (Exploratory behavior)

Turn the model's head towards the m-th egocentric direction. The output of the i-th neuron of the M-layer, Y_M^i, is 1 if $i=m$, and 0 otherwise. The i-th neuron of the F-layer is given by following equations:

$$U_F^i = \sum_j w_{FV}^{ij} \cdot Y_V^j + \alpha \cdot \sum_{j,k} w_{FFM}^{ijk} \cdot Y_F^j \cdot Y_M^k, \quad (1)$$

$$Y_F^i = \begin{cases} 1, & i = i_a, U_F^{i_a} = \max_{i \neq i_0}\left(U_F^i\right) \\ 0, & others, \end{cases} \quad (2)$$

where U_F^i is the membrane potential of the neuron i of the F-layer. Y_F^i is the output of the neuron i of the F-layer. Y_V^j is the output of the neuron j of the V-layer, $Y_V^j=1$ if the model looks the j-th view and $Y_V^j=0$ otherwise, and α is a positive constant (= 0.7).

The second term of the right-hand side of the equation (1) is the multiplication of Y_F^j and Y_M^k. It does not necessarily need the multiplication instead of the addition, essentially. The multiplication of Y_F^j and Y_M^k guarantees that the cell that will be active next is also different, if the movement direction is different.

If $U_F^{i_a}$ is below the threshold value (<0.2), the F-layer causes neurogenesis as follows:

$$w_{FFM}^{ijk} = 0, \ i = i_a, \sum_{j,k} w_{FFM}^{i_a jk} = \min_i \sum_{j,k} w_{FFM}^{ijk}, \forall j, \forall k, \quad (3)$$

$$w_{FV}^{ij} = 0, \quad i = i_a, \sum_{j,k} w_{FFM}^{i_a jk} = \min_i \sum_{j,k} w_{FFM}^{ijk}, \forall j, \quad (4)$$

$$Y_F^i = \begin{cases} 1, & i = i_a, \\ 0, & others. \end{cases} \quad (5)$$

The equation (3) and (4) shows the extinction of connections. It can interpret as death of the cell. The equation (5) shows the spontaneous activation of the cell having no connection with other cells. It means the generation of a new cell. Changes in connections w_{FV}^{ij}, w_{FFM}^{ijk} are given by following equations:

$$\mathrm{d}\, w_{FV}^{ij} / \mathrm{d}t = k1 \cdot Y_F^i \cdot \left(Y_V^j - w_{FV}^{ij}\right), \quad (6)$$

$$\mathrm{d}\, w_{FFM}^{ijk} / \mathrm{d}t = k2 \cdot Y_F^i \cdot Y_M^k \cdot \left(Y_F^j - w_{FFM}^{ijk}\right) + k3 \cdot (0 - w_{FFM}^{ijk}), \quad (7)$$

where $k1$, $k2$ and $k3$ are positive constants where ($k1$, $k2$, $k3$) = (0.6, 0.2, 0.0001). The activity of the RW cell at time $t=0$, Y_{RW}^0, is given by the following equation:

$$Y_{RW}^0 = \max_i \left(w_{RF}^i \cdot Y_F^i\right), \quad (8)$$

where w_{RF}^i is the connection from the neuron i of the F-layer to the RW cell.

Step 2. Reflection mode (Prediction of a goal)

1. Time $t=1$.
2. The i-th neuron of the F-layer is given by following equations:

$$U_F^i = \sum_{j,k} w_{FFM}^{ijk} \cdot f\left(\mathrm{d}\, Y_F^j / \mathrm{d}t\right) \quad (9)$$

$$Y_F^i = \begin{cases} 1 - 0.01 \cdot t, & U_F^i > 0.3, \\ 0, & others, \end{cases} \quad (10)$$

where $f(x)=1$ if $x>0$, and $f(x)=0$ otherwise. The activity of the RW cell at time t, Y_{RW}^t, is given by the following equation:

$$Y_{RW}^t = \max_i (w_{RF}^i \cdot Y_F^i) \quad (11)$$

3. Add 1 to t. Repeat 2 of Step 2 while $t<10$.
4. The activity of the m-th neuron of the W-layer is given by following equation:

$$Y_W^m = \max_t \left(Y_{RW}^t \right) \quad (12)$$

Add 1 to m and repeat Step 1 and Step 2 until the model finishes looking around all views from the present position. When finished, reset m to 0.

Step 3. Movement

Obtain the egocentric direction mr which has the highest value of Y_W^i. If the model has acquired the coordinate system, it can successfully activate the RW cell and cells of the W-layer. The value of Y_W^{mr} is in inverse proportion to the distance from model's present position to the goal and the direction mr means the goal direction.

Change mr to a random direction if $Y_W^{mr}<0.6$ or impossible to move towards the mr-th direction. $Y_M^i=1$ if $i=mr$, and $Y_M^i=0$ otherwise. $Y_F^i=1$ if $i=i_0$, and $Y_F^i=0$ otherwise. Repeat the equation (1) and (2) once again and move towards the mr-th egocentric direction. Substitute i_a for i_0. Add mr to dr if $mr+dr<8$, add mr-8 to dr otherwise. Go back to Step 1 until the model reaching a goal or a maximum 10 times. When the model reaches the goal, finish the trial and change w_{RF}^i by the following equation:

$$\mathrm{d}w_{RF}^i/\mathrm{d}t = k4 \cdot Y_F^i \cdot \left(1 - w_{RF}^i\right) \quad (13)$$

where $k4$ is a positive constant (=1). If the model is placed into a novel environment, w_{FV}^{ij} are reseted to 0.

3 Computer experiments

3.1 Prediction of a shortcut in a novel environment

In this section, we demonstrate that the model can predict a shortcut in a novel environment after learning various routs in a previous environment except for a rout that has a same sequence with a tested path in the novel environment. First, the model moves randomly in the environment *I* shown in Fig.3a, but not follows a way of which movement pattern is identical with the Path *A*'s on the environment *II*. When the model reaches N in the environment *I* or moves for 10 steps, activation of neurons in the F-layer is resetted to the initial values and the model starts again from S in the environment *I* as a next trial. We prepared 7 experimental conditions as total trials in the environment *I*, 0, 5, 10, 15, 20, 25 and 30 trials. At each condition, 60 model rats, given different initial values, were tested.

Then the model is moved into the environment *II* and forced to run Path *A* at 10 times. When the model reaches G in the environment *II*, reward is given to the model. Then the model is placed in the starting point S of the environment *II* and tested which rout the model selects.

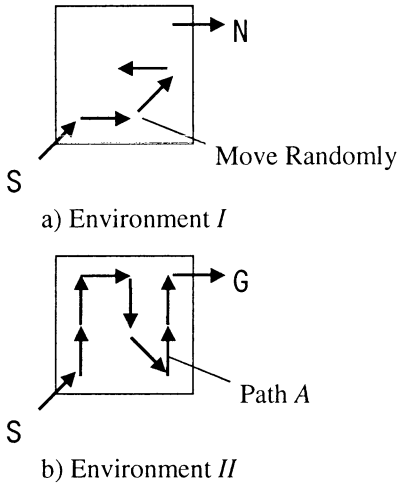


Fig.3. Schematic representation of the environment *I* and the environment *II*.

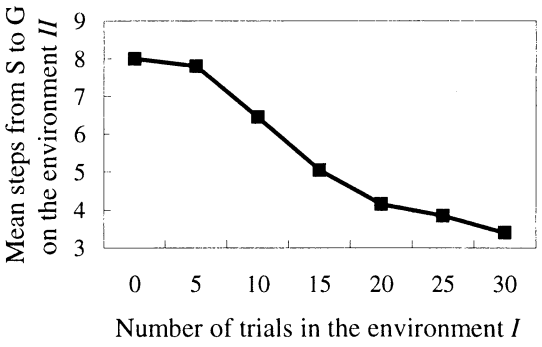


Fig.4. Mean steps of 60 models from S to G that the model had taken.

As a result, when no trial in the environment *I* had been given to the model, all models merely followed the Path *A*. After 30 trials in the environment *I*, almost all models took the shortest path in the environment *II*.

Notice that after the model fully learned various routs in the environment *I*, the model could generate the appropriate sequence while the same sequence with the Path *A* was inexperience, and could take a shortcut never experienced before. It means that the model acquired the coordinate system properly.

3.2 Emergence of the cognitive space

Probably, real rats acquire the coordinate systems through experiencing various environments. In this section, we gave 40 environments to the model in succession as shown in Fig.2. The model explores environments successively. First, the model moves randomly on an environment for 10 steps from the start position. The model repeats the random search until it reaches the goal. When the model reaches the goal, the model is placed into S on the environment again and tested that whether the model can take a shortcut at the next trial. If the model cannot take a shortcut, random

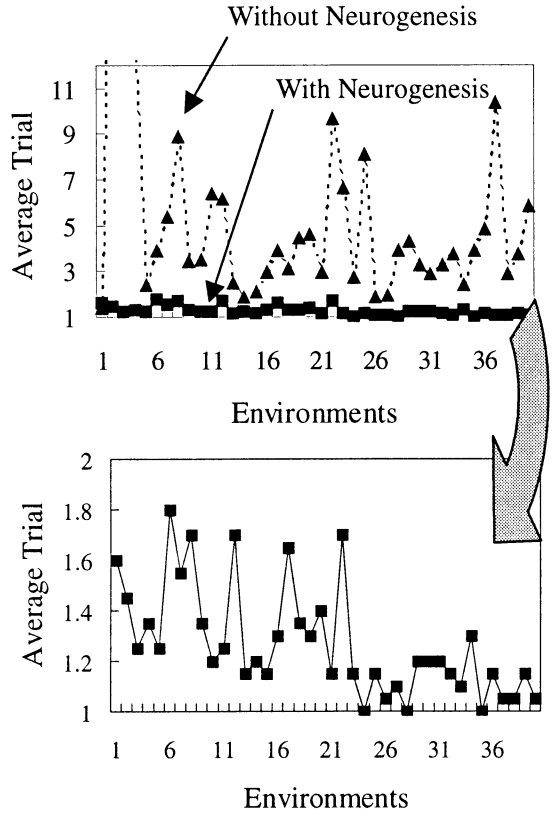


Fig. 5. Mean trials until the model can take the shortest path.

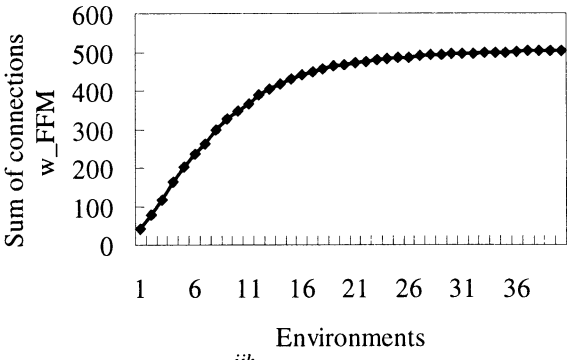


Fig. 6. Sum of w_{FFM}^{ijk} at each environment

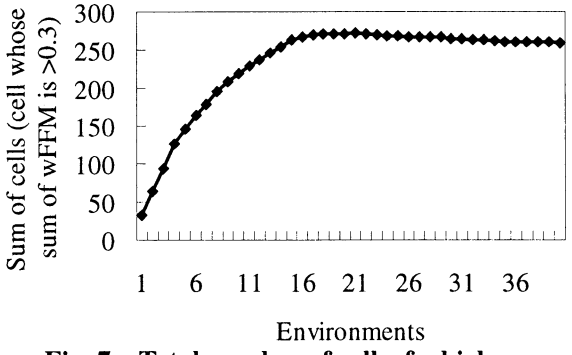


Fig. 7. Total number of cell$_i$ of which $\sum_{j,k} w_{FFM}^{ijk}$ is >0.3

search is repeated in the same environment. If it can take a shortest way, total trials taken in the environment are counted and the model is placed into S on the next environment.

40 models without the neurogenesis and 20 models with the neurogenesis are examined. As a result, each model with the neurogenesis was able to gradually take a shortcut in novel environments after one trial (Fig.5). Although Fig.6 shows increase and saturation of the sum of w_{FFM}^{ijk}, the number of cells relating to the network with high w_{FFM}^{ijk} (>0.3) has decreased in the last 20 environments (Fig.7). It means that, though the connections had dispersed in the beginning by the neurogenesis, it was integrated to cells those survived through learning several environments.

4 Summary

The presented model started from low ability to merely follow the path to the goal experienced just before. Then, the model acquired the algorithm for metric navigation by relating views of each environment to the network and avoiding contradicting connections with the neurogenesis. Such system — organizing universal rule to achieve its goal — would help to realize other biological process like acquisition of language.

References

[1] Tolman EC, Ritchie BF and Kalish D (1946), Studies in spatial learning. I. Orientation and the short-cut. *Journal of Experimental Psychology* 36:13-24

[2] Tolman EC (1948), Cognitive maps in rats and men. *Psychological Review* 55:189-208

[3] Trullier O, Sidney IW, Alain B and Meyer J-A (1997), Biologically based artificial navigation systems: Review and prospects. *Progress in Neurobiology* 51:483-544

[4] Trullier O and Meyer J-A (1997), Biomimetic navigation models and starategies in animats. *AI Communications* 10:79-92

[5] Bolles RC and Woods PJ (1964), The ontogeny of behaviour in the albino rat. *Animal Behaviour* 12: 427-441

[6] Raunch SL and Raskin LA (1984), Cholinergic mediation of spatial memory in the preweanling rat: Application of the radial arm maze paradigm. *Behavioral Neuroscience* 98:35-43

[7] Kempermann G and Gage FH (1999), New nerve cells for the adult brain. *Scientific American* 280(5):48-53

[8] Altman J and Das GD (1965), Post-natal origin of microneurones in the rat brain. *Nature* 207:953-956

[9] Kempermann G, Kuhn HG and Gage FH (1997), More hippocampal neurons in adult mice living in an enriched environment. *Nature* 386:493-495

[10] O'Keef J and Nadel L (1978), The hippocampus as a cognitive map. Clarendon Press, Oxford, U.K.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

A new method to improve generalization ability of neural networks

Baiquan Lu, Kotaro Hirasawa and Junichi Murata

Graduate School of Information Science and Electrical Engineering
Kyushu University, Hakozaki, Higashi-ku, Fukuoka 812-8581, Japan
Tel: (092) 642-3799 Fax: (092) 642-3962
E-mail: lu@terra.ees.kyushu-u.ac.jp

Abstract

In this paper, a new method to improve generalization abilities of neural networks is proposed that assures small error on non-training data as well as on training data. The method is to improve generalization abilities of neural networks by selecting optimum data from given training data, which best characterize the teaching function. The method can be employed for function approximation problems.

keywords : neural network learning, training data selection, hardness of learning, easiness of learning

1 Introduction

For improvement of generalization abilities of neural networks, usually there are three methods: regularization method, method using additive noise, and pruning method. Regularization method and pruning method are methods of constraining weights, while method using additive noise [1] is equivalent to regularization method in some situations. However these methods do not consider the relation between training data and test data. If characteristic of teaching function included in training data is different from that included in test data, even if the learning for training data is good, its generalization ability should not be good. For this, in the paper, a method is proposed to study how to select training data to improve generalization abilities. In function approximation problems, as for generalization abilities of neural networks, there are two problems: one is to select optimum data from given training data, which best characterize teaching function, and the other is to learn these data with small error. To solve these problems, we have to define a measure of easiness and hardness of leaning determined by the error measure. If the error measure after learning is large, the learning for these data said to be difficult, otherwise it is said to be easy. Because the large error after learning implies that the neural network has not well learned the characteristics of teaching function described by those training data, its generalization ability should not be good. Though the small error after learning implies that the neural network describes the characteristics of teaching function contained in those training data, its generalization ability is sometimes not good since the characteristics learned by the network is only part of characteristics of teaching function. For the large error after learning, we have to select training data which are easy for learning than before. For the small error after learning, we have to select training data which are difficult for learning than before. The learning is carried out initially with easy data then with hard data, like a human does, in order to select optimum data from given training data, which best characterize teaching function. In the learning choice of hard training data for the learning and choice of easy training data for the learning is done by using a BP method for small change of inputs. It is proved that this method is equivalent to regularization method for small change of inputs.

2 Learning method

In this section, we will give a learning method based on data with varying easiness/hardness. Consider a zero-sum, two-player game described by finite cost $E(W, X)$ of the form

$$E(W, X) = \sum_{i}^{n} (f(X(i)) - r(X(i)))^2 \qquad (1)$$

where $r(X(i))$ and $f(X(i))$ are desired and actual neural network outputs corresponding to the i_{th} training data, respectively. n is the number of training data, W a vector constituted by all the weights of neural network.

Aiming at making the cost function small in the worst training input in order to improve generalization ability of neural networks, the cost function is to be maximized by input vector X (player 2) and minimized by weights vector W (player 1), respectively. A necessary

and sufficient condition for the existence of saddle-point solution to a zero-sum game problem is given by the following Theorem.

Theorem 1: Consider a zero-sum, two-player game. Two policies X^*,W^* provide saddle-point solution (i.e., $E(W^*, X) \leq E(W^*, X^*) \leq E(W, X^*)$, for $\forall W$ and $\forall X$), if and only if the following holds

$$\min_W \max_X E(W, X) = \max_X \min_W E(W, X). \quad (2)$$

Equation (2) is often used to find the saddle-point solution. For example, starting with a given W, the learning is done in the order of $\max_X E(W, X) \rightarrow \min_W E(W, X) \rightarrow \max_X E(W, X) \cdots$, in other words, *min* operation and *max* operation are done alternately. Because BP methods have parallel calculative ability in some level, we use the following (similar to BP method) to find the saddle-point solution.

$$\dot{W} = -\eta_1 \frac{\partial E}{\partial W}$$
$$\dot{X} = +\eta_2 \frac{\partial E}{\partial X}$$

$$\begin{aligned} \dot{E}(W, X) &= (\frac{\partial E}{\partial W})^T \dot{W} + (\frac{\partial E}{\partial X})^T \dot{X} \\ &= -\eta_1 (\frac{\partial E}{\partial W})^T \frac{\partial E}{\partial W} + \eta_2 (\frac{\partial E}{\partial X})^T \frac{\partial E}{\partial X} \end{aligned} \quad (3)$$

From this, we know that learning of X make the error function $E(W, X)$ large and learning of W make the error function $E(W, X)$ small. Because W and X are adjusted together, the learning has to enter a local *minmax* $E(W, X)$ after learning. Now we use EHL method (easy and hard learning) to try to solve this problem. When the learning enter a local minimum for weights by BP method, if the error function is still large, learning for those training data is difficult, we have to choose easy data from all of training data to learn again. If error function is small, we have to choose hard data from all of training data to learn again.

We call X_2 easier training data than X_1 if $E(W, X_1) > E(W, X_2)$, otherwise harder training data than X_1.

For choice of harder training data for the learning, we use the following

$$\Delta X = X_2 - X_1 = +\eta_3 \frac{\partial E}{\partial X_1}. \quad (4)$$

If $X_2 = X_1 + \Delta X$ is not a member of training set for a given training set, we can use a member of training set closest to X_2.

For choice of easier training data for the learning, we use the following

$$\Delta X = X_2 - X_1 = -\eta_4 \frac{\partial E}{\partial X_1}. \quad (5)$$

If $X_2 = X_1 + \Delta X$ is not a member of training set for a given training set, we can use a member of training set closest to X_2.

Thus, equation (3) becomes the following,

$$\begin{aligned} \Delta E(W, X) &= (\frac{\partial E}{\partial W})^T \Delta W + (\frac{\partial E}{\partial X})^T \Delta X + \cdots \\ &= -\eta_1 (\frac{\partial E}{\partial W})^T \frac{\partial E}{\partial W} + \eta_3 (\frac{\partial E}{\partial X})^T \frac{\partial E}{\partial X} + \cdots \end{aligned} \quad (6)$$

We know that adjusting of weights of neural networks by BP method is converges if learning rate is small, otherwise it is over-learning or it is not convergent. Similar to this, adjusting of inputs variable X is convergent if learning rate is small, otherwise it is over-learning (this is contrary to BP method, that is, though the error function will become small, but it is bounded below, thus, η_3 and η_4 can be large positive numbers), in other words, adjusting of inputs variable X is almost everywhere converges for any learning rate. This is import for how to find a global *minmax* of the error function in next.

If $(\frac{\partial E}{\partial W})^T \approx 0$, $(\frac{\partial E}{\partial X})^T \neq 0$ and the error function $E(W, X)$ is still large, $\Delta E(W, X) > 0$ for small learning rate η_3, thus, the learning for future is not good, this implies that learning for those training data is difficult. For this, we have to use (5) to select easier training data than before and learning again, thus we have the following

$$\Delta E(W, X) = -\eta_1 (\frac{\partial E}{\partial W})^T \frac{\partial E}{\partial W} - \eta_4 (\frac{\partial E}{\partial X})^T \frac{\partial E}{\partial X} + \cdots \quad (7)$$

We use this to learning again until the error function is small, then we use (6) to select harder training data and learning until the error function is small, thus a local *minmax* of the error function is found and the error function is small near region of the local *minmax* of the error function. For that the error function is also small for large region, we have to find other local *minmax* of $E(W, X)$. Because the error function is small when $(\frac{\partial E}{\partial W})^T \approx 0$, we can use equation (7) and select large learning rate η_4 in order to try to find large the error function value than before. Thus the learning enter region of other local *minmax* of $E(W, X)$ until the global *minmax* of the error function is found.

If ΔX is small, this method is equivalent to regularization method. At X_1, we have the following by Taylor expansion

$$\begin{aligned} E(W, X) = E(W, X_1) + \frac{\partial E}{\partial X} \Delta X + \cdots \\ = U + R \end{aligned}$$

where $U = E(W, X_1)$ is usual term for learning of weights of neural networks, while R is regularization

term for learning of weights of neural networks. This method is different from method using additive noise in back-propagation training. The method using additive noise in back-propagation training is to use random to get change of input variable, while our method is to get change of input variable by selecting training data in mean of *minmax*, such as when we use (4), because ΔX is small, we have $E(W,X) > E(W,X_1)$, this implies that X is harder than X_1 for learning, while when we use (5), because ΔX is small, we have $E(W,X) < E(W,X_1)$, this implies that X is easier than X_1 for learning.
Because $\frac{\partial E}{\partial X} \approx 0$ after learning, $E(W,X)$ is not sensitive to input X, that is, it is robust for input X at this point.
Since for calculation of ΔX, we have to know partial derivatives of teaching function with respect to input, and data with easy or hard than that of now are much, we can use the following to approximate partial derivatives of teaching function with respect to input.

$$\frac{\partial r(X(i)}{\partial x_j} \approx \frac{r(X(i)) - r(X(i+1))}{x_j(i) - x_j(i+1)},$$
$$i = 1, 2, \cdots, n, j = 1, 2, \cdots, m,$$
$$X = [x_1, x_2, \cdots, x_m].$$

where m is the total number of input.
This leads us to the following algorithm.
Algorithm

Step 1) Given training data, weights are trained by using BP method for part of training data until the learning enter local minmxa.

Step 2) If error function is still large, choose easy data from all of training data to learn again. If error function is small, choose hard data from all of training data to learn again.

Step 3) Weights are trained by using BP method for those data until the learning enter local minima. If the error function is also small for non training data, stop, otherwise go to Step 2).

3 Example

In this example, the desirable function is assumed to be of the form [2]:

$$f(x) = 0.2 + 0.8(x + 0.7\sin(2\pi x)). \quad (8)$$

The training data are taken at every interval of 0.16, for $0 < x < 2$, thus we have 12 data points

$$(0.16, f(0.16)), (0.32, f(0.22)), \cdots, (1.92, f(1.92)).$$

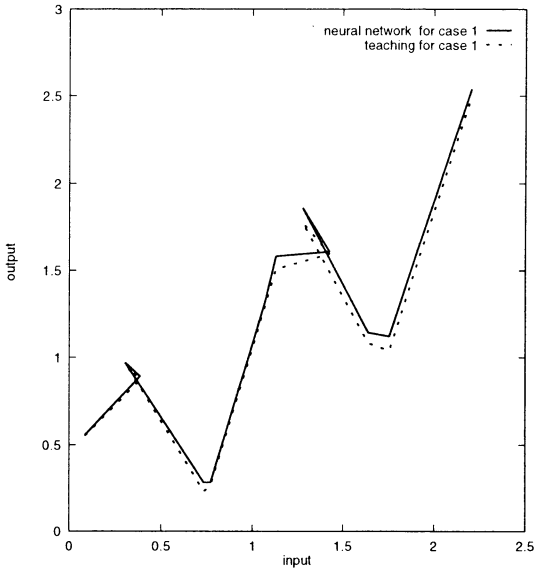


Figure 1: Teaching function and neural networks after learning on training data

The neural network structure is chosen as 1-40-1, the node function is $\varphi(x) = (1 - e^{-x})/(1 + e^{-x})$, initial values of weights connecting 1st to 2nd and 2nd to 3rd layers are generated from the uniform distribution (-1.5,1.5) and (-1.0,1.0), respectively. We take 76 data points generated from the uniform distribution (0,2.0) to verify the generalization ability of the network. The learning is done for 100000 iterations with learning rate 0.0002, and the results are compared with the ordinary BP learning with learning rate 0.015 and momentum term coefficient 0.001. The outputs of neural networks after learning are shown in Fig.1. The outputs of neural networks for non-training data are shown in Fig.2, the average errors of our method and ordinary BP learning on non-training data are 0.160138 and 0.30444, respectively. This implies that generalization ability of neural network trained by our method is better. The frequency of training data are shown in Fig.3. From Fig.3, we can see that all of training data to be chosen after learning are in sharp region of teaching function. Because the characteristic of teaching function can be well express by this part, this implies that neural networks can get the characteristic of teaching function by using our method. We can see from the plots that learning of neural network by the proposed method is very useful for improving generalization ability.

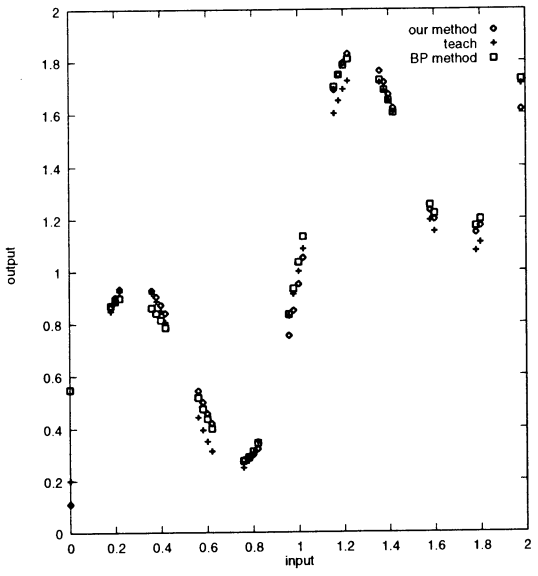


Figure 2: Networks outputs and teaching function on non-training data

4 Conclusion

A new method to improve generalization abilities of neural networks has been proposed. Its learning is carried out by selecting optimum data from given training data, which best characterize teaching function. At the same time, the learning is carried out initially with easy data then with hard data like a human does. Its effectiveness and applicability have been demonstrated by examples.

References

[1] Lasse Holmstrom and Petri Koistinen, "Using additive noise in back-propagation training", *IEEE TRANSACTIONS ON NEURAL NETWORKS*, vol. 3, no. 1, pp. 24–38, 1992.

[2] Brijesh Verma, "Fast training of multilayer perceptrons", *IEEE TRANSACTIONS ON NEURAL NETWORKS*, vol. 8, no. 6, pp. 1314–1320, 1997.

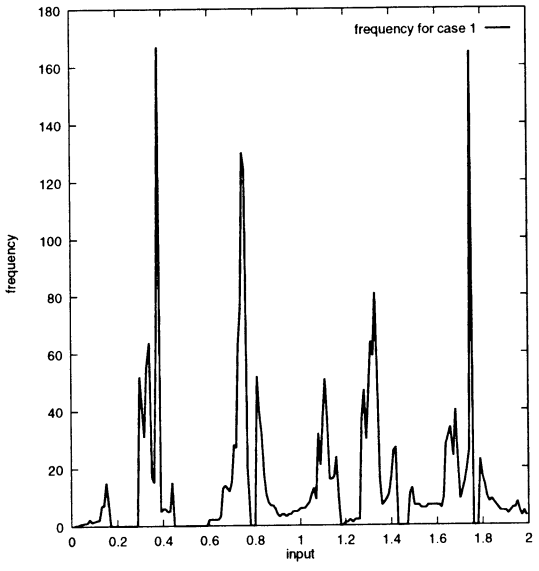


Figure 3: Learning number on input points

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Random Iterated Neural Networks as Dynamical Systems Universal Approximators

Fernando Niño
German Hernandez
Dept. of Mathematical Sciences
University of Memphis and
Universidad Nacional de Colombia

Fernanda Botelho
Anthony Quas
Dept. of Mathematical Sciences
University of Memphis
Memphis, TN 38152

Abstract

This paper shows that random iterated neural networks are universal asymptotic approximators of dynamical systems that have a physical measure supported on a compact global attractor.

1 Introduction

Over the past decade, there has been an increasing interest in recurrent neural networks that can approximate the behavior of dynamical systems. The motivation for this interest is their potential application in control theory, identification and simulation of nonlinear systems, as well as in forecasting of nonlinear time series. Nonlinear systems and nonlinear time series appear in virtually every field of science, such as economics, physics, biology and engineering.

Several results have been proved for different types of recurrent networks. For instance, a class of continuous time recurrent neural networks (known as globally asymptotically stable neural networks) was shown to converge to an unique equilibrium [10]; and these networks are capable of approximating any finite time trajectory of a given n-dimensional continuous dynamical system.

A *random iterated neural network* [9] [13] implements a discrete random dynamical system [3] defined by an iterated random function system [5]. We prove that these networks are universal asymptotic approximators for dynamical systems with a compact global attractor and a unique invariant measure, up to any degree of accuracy. We will assume that the invariant measure is a physical measure [6].

The rest of this paper is organized as follows: in section 2 some basic definitions from dynamical systems are reviewed. In section 3, random iterated neural network are described. Finally, the proof of the universality of RINN is presented in section 4.

2 Preliminaries

2.1 Dynamical Systems

Here, we consider a *discrete dynamical system* as a pair $\langle X, f\rangle$ where X is a compact metric space, called *phase space*, and f is a map on X, called *dynamics*. Study of dynamical systems usually concentrates on the long-term behavior of the orbits $\{f^n(x)\}_{n=0}^{\infty}$ for almost every $x \in X$. We are interested in systems with only one invariant set A_f ($f(A_f) = A_f$) and whose global dynamics converges to it. In this case, such invariant set is called the global attractor of the system.

2.1.1 Global Attractor

Intuitively, an attractor is a set in which every orbit of the system accumulates to. Attracting fixed points and periodic orbits are examples of attractors, but also attractors with complicated geometry can be found [1].

An attractor is an attracting set that is irreducible in the weakest sense, i.e., it has a dense orbit. For instance, the union of two disjoint attractors is not an attractor. A *global attractor* is an attractor whose basin of attraction is X. Here, we will consider only dynamical systems with *compact global attractor* [6].

2.1.2 Asymptotic Equivalence

Let us consider a dynamical system $\langle X, f\rangle$ in which the phase space X is a compact metric space and f is a continuous (or piecewise continuous) mapping. Under these assumptions, we can study the time evolution of probability measures under the dynamics of the system. We are therefore interested in the set of Borel probability measures $\mathbb{M}(X)$ of the system and are looking for the measures describing the long-term behavior of the system.

Let $\langle X, f\rangle$ and $\langle X, g\rangle$ be dynamical systems with compact global attractors A_f and A_g and physiscal measures π_f and π_g, respectively. The systems $\langle X, f\rangle$ and $\langle X, g\rangle$ are *asymptotically equivalent* if and only if

$A_f = A_g$ and $\pi_f = \pi_g$. Additionally, given $\varepsilon > 0$, we say that $\langle X, f\rangle$ is an ε- *asymptotic approximation* of $\langle X, g\rangle$ if and only if the Hausdorff distance h between their universal attracting sets is less than or equal to ε, and also the Hutchinson distance H [11] between their physiscal measures is less than or equal to ε, i.e.,

$$h\left(A_f, A_g\right) \leq \varepsilon \text{ and } H\left(\pi_f, \pi_g\right) \leq \varepsilon.$$

2.2 Random Iterated Function Systems

An *Iterated Function System* (IFS) consists of a set of maps acting on a metric space (see [11], [15]). Here, we will consider hyperbolic IFS, i.e., finite collections of contractive maps on a compact metric space ([4]). An IFS is denoted by $\{X; f_1, f_2, ..., f_k\}$, where (X, d) is a compact metric space and each $f_i : X \to X$ is a contractive map on X with contractivity factor ζ_i. The value $\zeta = \max\{\zeta_i : i = 1, ..., k\}$ is called the *contractivity factor of the IFS.*

A *random iterated function system (*also called *iterated random function* or *probabilistic iterated function system*) ([5] [15]) is an IFS with a set of probabilities $\{p_i \mid i = 1, 2, ..., k\}$ associated to the maps, where $p_i > 0$ and $p_1 + p_2 + ... + p_k = 1$.

A random iterated function system (RIFS) will be denoted as $\{X; f_1, f_2, ..., f_k; p_1, p_2, ..., p_k\}$. A RIFS defines a random discrete dynamical system $\langle X, F\rangle$ with

$$F(x) = \begin{cases} f_1(x) & \text{with probability} \quad p_1 \\ f_2(x) & \text{w.p.} \quad p_2 \\ & \vdots \\ f_k(x) & \text{w.p.} \quad p_k \end{cases}$$

which has a compact global attractor A_F, and a unique invariant meausre π_F supported on the attractor; this measure is the attractor visiting probability measure. Besides, an orbit of any arbitrary point x_0 is a random walk $\{x_0, x_1, ...\}$ on X. Hutchinson's theorem [11] establishes that

$$\omega_F(x_o) = \bigcap_{n \geq 0} \overline{\bigcup_{r=n}^{\infty} F^r\left(x_o\right)}$$

converges, in the sense of Hausdorff distance, to the global attractor, i.e., $\omega_F(x_o) \stackrel{h}{\to} A_F$ for every $x_0 \in X$.

For any particular assignment of probabilities $p_1, p_2, ..., p_k$ the time evolution of probability measures by the RIFS is described by Markov operator of the form,

$$\mathcal{M}(v) = \sum_{i=1}^{k} p_i v \circ f_i^{-1} \text{ for } v \in \mathbb{M}(X).$$

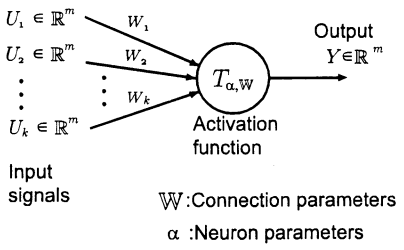


Figure 1: Hyperneuron.

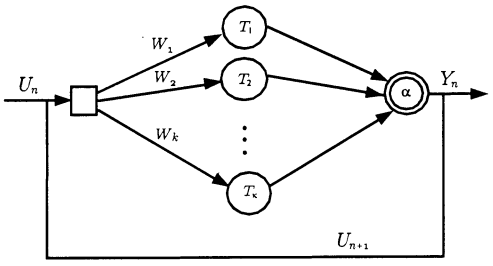


Figure 2: Architecture of RINN

Figure 3: Random iterated neural network. Input layer: a hyperneuron that distributes the input to the second layer. Hidden layer: k contractive hyperneurons. Ouput layer: a probabilistic hyperneuron with neuron parameter $\alpha = (p_1, p_2, ..., p_k)$.

Elton's ergodic theorem [7] stablishes that there exists a unique invariant measure supported in the atractor and this measure is also a physical measure of the RIFS.

3 Random Iterated Neural Networks

A Random Iterated Neural Network (RINN) is a three-layered recurrent discrete neural network that implements an iterated random function and is composed of units called hyperneurons, explained next.

A *hyperneuron* [9] (see Figure 1 is an extension of the model of a neuron in which the inputs, activation and outputs are vector valued. A *contractive hyperneuron* is a hyperneuron that implements a contractive map. On the other hand, a *probabilistic hyperneuron* is a special type of hyperneuron that implements a random map.

A RINN is recurrent neural with three layers (see figure 3) that implements a random discrete dynamical

system $\langle \mathbb{R}^m, F \rangle$ given by

$$F(U) = \begin{cases} T_1(U) & \text{with probability} \quad p_1 \\ T_2(U) & \text{w.p.} \qquad\qquad p_2 \\ \qquad \vdots & \\ T_k(U) & \text{w.p.} \qquad\qquad p_k \end{cases}$$

associated to a random iterated function system $\{\mathbb{R}^m; T_1, T_2, ..., T_k; p_1, p_2, ..., p_k\}$.

4 Universality

Throughout the rest of this paper, we consider dynamical systems on the unit hypercube $D = [0,1]^m$ with the Euclidean distance. In this section, we prove that random iterated neural networks are universal asymptotic approximators of dynamical systems on D with compact global attractor and unique invariant attractor visiting probability measure. This means that given a dynamical systems on D, which has compact global attractor and unique invariant attractor visiting probability measure, and given $\varepsilon > 0$, there exists a RINN whose dynamics is an ε-asymptotic approximation of the dynamics of the system.

Let $\mathbb{H}(D)$ be the collection of all compact non-empty subsets of D. $\mathbb{H}(D)$ with the Hausdorff distance h, is a complete metric space (see [4]). Now, let $\mathbb{F}(D)$ be the collection of finite subsets of D.

Proposition $\mathbb{F}(D)$ is dense in $(\mathbb{H}(D), h)$.

Proof. Let $A \subset D$ be a non-empty compact set. Clearly, A is closed and bounded. Let ε be a positive number ($\varepsilon > 0$). Let us consider the grid G^ε of size ε on D, defined as follows,

$$G^\varepsilon = \{\mathbf{x} = (x_1, x_2, \cdots, x_m) \in D \mid x_j = i\varepsilon \text{ for } i \in \{0, 1, \cdots, \left[\tfrac{1}{\varepsilon}\right]\}, \text{ for } j = 0, 1, ..., m\}.$$

Figure 4 shows G^ε for $D = [0,1]^3$, the three-dimensional hypercube. Let $B_\varepsilon(\mathbf{z}_i)$ be the m-dimensional hypercube centered at the point $\mathbf{z}_i == (z_{i1}, z_{i2}, \cdots, z_{im})$, and such that each of its sides has length ε, as follows, $B_\varepsilon(z_i) = \left(\prod_{j=1}^m \left[z_{ij} - \frac{\varepsilon}{2}, z_{ij} + \frac{\varepsilon}{2}\right)\right) \cap D$. Let us define the *$\varepsilon$-finite approximation* of A as $A^\varepsilon = \{\mathbf{x} \in G^\varepsilon \mid B_\varepsilon(\mathbf{x}) \cap A \neq \emptyset\}$. Clearly, A^ε is a finite set such that $h(A, A^\varepsilon) < \varepsilon$, which completes the proof.

Approximation Theorem Let $\langle D, g \rangle$ be an autonomous discrete dynamical system with compact global attractor A_g and physical measure π_g. Then for any $0 < \varepsilon < 1$, there exists a RINN that is an ε-asymptotic approximation of $\langle D, g \rangle$.

Proof. Let ρ be a positive constant, $\rho = \frac{\varepsilon}{\sqrt{m}}$. Let $A_g^\rho \subset D$ be the ρ-finite approximation

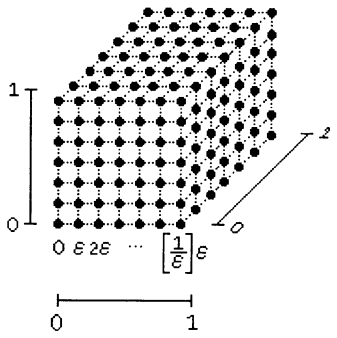


Figure 4: Grid G_ε for $D = [0,1]^3$.

of set A, say $A_g^\rho = \{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_k\}$ with $\mathbf{z}_i = (z_{i1}, z_{i2}, \cdots, z_{im})$. Now, let us consider the constant functions $f_i : D \to D$, for $i = 1, 2, ..., k$, defined as $f_i(x) = \mathbf{z}_i$ for all $x \in D$. Clearly, each f_i is a contractive map on D.

Let us define the probabilities $p_1, p_2, ..., p_k$ as

$$p_i = \pi_g\left(B_\rho(\mathbf{z}_i)\right), \text{ for } i = 1, 2, ..., k,$$

the probability that the system visits the set $B_\rho(\mathbf{z}_i)$. Without loss of generality, let us assume that $p_i > 0$ for all $i = 1, 2, ..., k$, and also $p_1 + p_2 + ... + p_k = 1$. Therefore, these functions and probabilities define the random iterated function system $\{D; f_1, f_2, ..., f_k; p_1, p_2, ..., p_k\}$. This RIFS determines a dynamical system $\langle D, F \rangle$ whose global attractor is A_g^ρ. Clearly, $h(A_g^\rho, A_g) \leq \rho$. Additionally, $\langle D, F \rangle$ has a unique invariant measure

$$\pi_F = \sum_{j=i}^{k} p_i \delta_{\mathbf{z}_i}$$

with $\delta_{\mathbf{z}_i}$ the measure concentrated in the point $\mathbf{z}_i$, supported on the attractor. Now, for any contractive function $\varphi : D \to \mathbb{R}$ we have that

$$\int_{B_i} \varphi \, d\pi_F = p_i \varphi(\mathbf{z}_i),$$

where $B_i = B_\rho(\mathbf{z}_i)$.

On the other hand, we have that

$$\int_{B_i} \varphi \, d\pi_g \leq p_i \left(\varphi(\mathbf{z}_i) + \frac{\rho}{2}\sqrt{m}\right) < p_i \left(\varphi(\mathbf{z}_i) + \rho\sqrt{m}\right)$$

then

$$\int_{B_i} \varphi \, d\pi_g - \int_{B_i} \varphi \, d\pi_F < p_i \rho \sqrt{m}$$

thus

$$\int_D \varphi \, d\pi_g - \int_D \varphi \, d\pi_F < \rho\sqrt{m} = \varepsilon,$$

i.e., $H(\pi_F, \pi_g) < \varepsilon$, with H the Hutchinson distance.

Therefore, if $\mathbf{N}$ is a random iterated neural network that implements $\langle D, F\rangle$, then by proposition 1 we have that $h(A_F, A_g) < \rho = \frac{\varepsilon}{\sqrt{m}} \leq \varepsilon$; besides, $H(\pi_F, \pi_g) < \varepsilon$. Therefore, $\mathbf{N}$ is an ε-asymptotic approximation of $\langle D, g\rangle$.

5 Summary

This paper showed that given a discrete dynamical system, with compact global attractor and physical measure, it is always possible to find an RINN, whose asymptotic behavior arbitrarily approximates the asymptotic behavior of the given dynamical system. Additionally, the approximation theorem gives the explicit construction of the RINN.

The results presented here have applications in fractal image compression and time series prediction. For instance, Theorem 2 gives a explicit solution to the inverse problem from fractal compression. On the other hand, an RINN may be used to approximate the behavior of a chaotic dynamical system; and therefore these results can be applied in the prediction of chaotic time series.

Acknowledgements

This work was supported in part by a grant from COLCIENCIAS to F. Niño and G. Hernandez

References

[1] R.H. Abraham, L. Gardini, and C. Mira. *Chaos in Discrete Dynamical Systems.* Springer Verlag, 1997.

[2] S. Amari. Characteristics of random nets of of analog neuron-like elements. *IEEE transactions on Systems, Man and Cybernetics*, 2:643–653, 1972.

[3] L. Arnold. *Random Dynamical Systems.* Springer Verlag, 1998.

[4] M. Barnsley. *Fractals Everywhere.* Academic Press, 1988.

[5] P. Diaconis and D. Freedman. Iterated random fucntions. *SIAM Review*, 41(1), 1999.

[6] E. Eckman and D. Ruelle. Ergodic theory of chaos and strange attractors. *Rev. Mod. Phys*, 57:617, 1985.

[7] J. Elton. An ergodic theorem for iterated maps. *Ergodic Theory and Dynamical Systems*, 7:481–488, 1987.

[8] K. Funahashi and Y. Nakamura. Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, 6:801–806, 1993.

[9] G. Hernandez, F. Niño, F. Botelho, and A. Quas. Random iterated neural networks: asymptotic behavior. In *Smart Engineering Systems: Neural Networks, Fuzzy Logic, Evolutionary Programming, and Rough Sets, C.H. Dagli et al (eds)*, volume 9. ASME press, 1999.

[10] M. Hirsh. Convergent activation of dynamics in continuous time neural networks. *Neural Networks*, 2:331–349, 1989.

[11] J.E. Hutchinson. Fractals and self-similarity. *Indiana Univ. Math. J.*, 30(5), 1981.

[12] E. Levin. Hidden control neural architecture modelling of nonlinear time varying systems and its applications. *IEEE Transactions on Neural Networks*, 4:109–116, 1993.

[13] F. Niño, G. Hernandez, and D. Dasgupta. Evolutionary design of random iterated neural networks. In *Smart Engineering Systems: Neural Networks, Fuzzy Logic, Evolutionary Programming, and Rough Sets. C.H. Dagli et al (eds)*, volume 9. ASME press, 1999.

[14] A. Quas. On representation of Markov chains by random smooth maps. *Bull. London math. Soc.*, 23:487–492, 1991.

[15] K. Shelton. Introduction to iterated function systems. *preprint University of North Carolina*, 1997.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

Reinforcement Learning with Autonomous State Space Construction using Unsupervised Clustering Method

K. Yamada, M.Svinin, and K. Ueda
Mechanical Engineering Department
Kobe University
Rokko-dai, Nada-ku, Kobe 657-8501, Japan

Abstract

This paper deals with reinforcement learning in the continuous state and action spaces. A classifier system with new mechanisms of action selection and reproduction is proposed. The system makes use of an unsupervised clustering method of dynamic segmentation the continuous state space. As learning progresses, the continuous state space is segmented by Bayes boundaries. The proposed system is tested under simulation for an autonomous mobile robot. It is shown that the learning performance can be essentially improved if clustering of the production rules is employed.

1 Introduction

One of the main problems in reinforcement learning is the construction of a mapping from the state space to the action space of the robot. The problem becomes complicated when the action space and the sensor space are continuous, as the discretization density directly affects learning performance. A coarse discrete state space gives rise to "incomplete perception", "hidden state" or "perceptual aliasing"[1]. On the other hand, a detailed discrete state space may lead to increasing the leaning time. To overcome this problem, it is necessary to segment continuous state space adaptively.

In our previous research, we have proposed classifier system[2] that was able to learn the continuous mapping from state space to action space. The system was tested for navigation tasks for the Khepera mobile robot and for the sensory-motor coordination of a multi-legged walking machine[3]. In our experience, the learning procedure, if not properly organized, may take a long time to converge since the robot has to explore a wide state space. To improve the learning performance, basic mechanisms of the classifier system should be carefully designed.

In this paper, we develop an unsupervised clustering method to segment the continuous state space. The proposed method employs learning vector quantization (LVQ[4]) and utilizes the Bayes decision theory to segment the continuous state space. The method is integrated into a learning classifier system. As learning progresses, the robot divides the continuous state space into the clusters.

2 Basic concept

In this section, we outline the learning procedure based on the use of a rule-based system. When a new sensory input is registered, theoretically, the control system of the robot should generate a new rule that consists of the current sensory input and the action to be executed. However, if a similar rule is already presented in the state space, there is no need to generate a new rule. To be able to classify similar rules, we utilize Bayes decision theory for comparing the sensory input with the states of the memorized rules.

Bayes decision theory is one of the well-known methods of clustering input data. In Bayes decision theory, when the prior probability $P(\boldsymbol{x})$ of the cluster C_i and the probabilistic density function $p(\boldsymbol{x}|C_i)$ are known in advance, one can determine the optimal cluster C_m to which input data $\boldsymbol{x}$ belong by the following formula:

$$P(\boldsymbol{x})_m p(\boldsymbol{x}|C_m) = \max_i \{P(\boldsymbol{x})_i p(\boldsymbol{x}|C_i)\}. \quad (1)$$

Assuming that the probabilistic density function of each cluster is given by the normal distribution, the covariance matrix and the mean of each cluster can be estimated step by step.

Bayes decision theory estimates the loss of misclassification of the sensory input. In our system, the action rules are associated with clusters, and the clusters are segmented by the Bayes boundaries. Basically, the

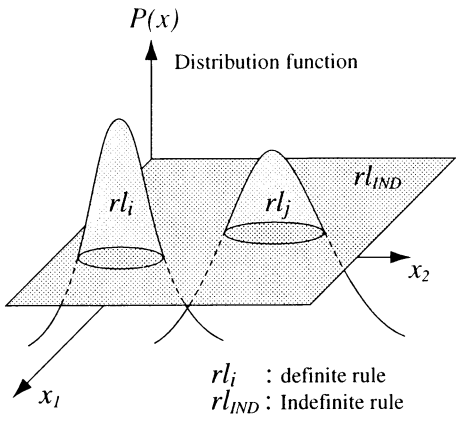


Figure 1: Distribution function.

robot selects one of the rules with smallest estimated losses and executes the corresponding action. If the action gets no punishment, the selected rules is placed near the sensory input and its utility increases. On the other hand, if the selected rule gets a punishment its utility decreases. In our system, the robot can not only generate and cluster new rules, but also can remove the rules whose fitness is under a certain threshold.

2.1 Structure of the Action Rules

Let n_s be the number of sensors a robot is equipped with, and $\boldsymbol{x} = \{x_1, \ldots, x_{n_s}\}^T$ be the sensory input of the robot. The system operates on a set of action rules, R. An individual rule can be associated with a cluster in the state space where the rule's action is appropriate. Formally, a rule $r \in R$ is defined as follows:

$$r :=< \boldsymbol{v}, u, a, f, \boldsymbol{\Sigma} > \qquad (2)$$

where $\boldsymbol{v} = \{v_1, \ldots, v_{n_s}\}^T$ is the state vector associated with and memorized in the rule r, u is the utility of the rule, a is the action corresponding to the rule r, f is the prior probability of the rule, $\boldsymbol{\Sigma}$is a covariance matrix of the rule's cluster. Note that the state vector $\boldsymbol{v}$ is the mean value of the rule's cluster,

If $\boldsymbol{v}$ matches in some sense the current sensory input $\boldsymbol{x}$, the rule r becomes active and can trigger its action a. The action rules defined as in (2) are called definite. They are represented by the normal distribution functions (Fig.1). In addition to the definite rules, the set R contains also an indefinite rule that is used for the exploration of the state space. The distribution function of indefinite rule r_{IND} is uniform and represented by a plane. If indefinite rule is selected, its action is determined randomly. Indefinite rule produces new rule which memorizes the observed sensory input and executed action.

In the beginning R consist of only one indefinite rule with initially assigned utility u_0. As learning progresses, the total number of rules in R, n_r, varies by reproduction and extinction.

2.2 Action selection

The rules in R compete with each other for the right to trigger their actions. The selection procedure is organized as follows. Given the probability density function of the i-th rule's cluster,

$$p_i(\boldsymbol{x}) = \frac{1}{(2\pi)^{\frac{n_s}{2}} |\boldsymbol{\Sigma}_i|^{\frac{1}{2}}} \cdot \exp\left\{ \frac{-1}{2} (\boldsymbol{x} - \boldsymbol{v}_i)^T \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{x} - \boldsymbol{v}_i) \right\}, \qquad (3)$$

and the estimated value of f_i, the risk of misclassification of the sensory input $\boldsymbol{x}$ into other clusters can be defined as

$$\begin{aligned} g_i &= -\log(f_i \cdot p_i) \\ &= \frac{1}{2} (\boldsymbol{x} - \boldsymbol{v}_i)^T \boldsymbol{\Sigma}_i^{-1} (\boldsymbol{x} - \boldsymbol{v}_i) \\ &\quad - \log\left\{ \frac{1}{(2\pi)^{\frac{n_s}{2}} |\boldsymbol{\Sigma}_i|^{\frac{1}{2}}} \right\} - \log f_i \end{aligned} \qquad (4)$$

Next, n_a active rules with the minimal values of g_i are selected to participate in the competition. The winner rule is defined randomly with the probability given by the weighted Boltzmann distribution:

$$P(r_j) = \frac{\exp(u_j / T_m)}{\sum_{k=1}^{n_a} \exp(u_k / T_m)} \qquad (5)$$

where T_m is a constant.

2.3 Temporal credit assignment

The utilities of the rules are updated every time after the winner executes its action. The mechanism of utility adjustment consists of four parts.

1. Direct payoff distribution The direct payoff P is given to the winner rules only in specific states. There are two types of payoff: reward ($P > 0$) and punishment ($P < 0$). The payoff is spreading back along the sequence of the rules triggered their actions (i.e., to the current and previous winners) with the discount rate $\gamma > 0$: $u_w^{(k)} \leftarrow + u_w^{(k)} + \gamma^k P, k = 0 \ldots N$, where N is the depth of the winners chain. Here, we denote the

parent of r_w as $r_w^{(1)}$, the parent of $r_w^{(1)}$ as $r_w^{(2)}$ and so on.

2. "Bucket brigade" strategy. The current winner r_w hands over a part of its utility, Δu, back to the previous winner, $r_w^{(1)}$: $u_w^{(1)} \leftarrow u_w^{(1)} + \Delta u$, where $\Delta u = \kappa(u_w - u_w^{(1)})$, $\kappa \in [0,1]$ if $u_w > u_w^{(1)}$ and $\Delta u = 0$, otherwise.

3. Taxation. Whenever a definite rule r_w triggers its action, its utility is updated as $u_w \leftarrow (1 - cf)u_w$.

4. Evaporation. All the rules reduce their utilities at the evaporation rate $\eta < 1$ when the robot reaches the goal state: $u_w \leftarrow \eta u_w$. The rules decreasing their utility below the threshold u min are removed.

2.4 Modification and Reproduction

The modification and reproduction phase is performed if the action triggered by the winner rule r_w has not been punished. If the winner rule is the indefinite rule a new rule, memorizing the current sensory input and the executed action, is produced. The parameters of the new rule are set as

$$\boldsymbol{v}_c = \boldsymbol{x}, \quad \boldsymbol{\Sigma}_c = \sigma \boldsymbol{I}, \quad a_c = a_w, \quad u_c = u_0, \tag{6}$$

where σ is a constant.

On the other hand, if the winner rule r_w is definite its parameters are modified as follows:

$$\boldsymbol{v}_w \leftarrow \boldsymbol{v}_w + \alpha \boldsymbol{\Sigma}_w^{-1}(\boldsymbol{x} - \boldsymbol{v}_w), \tag{7}$$

$$f_w \leftarrow f_w + \alpha(1 - f_w), \tag{8}$$

$$\boldsymbol{\Sigma}_w \leftarrow \boldsymbol{\Sigma}_w + \alpha[(\boldsymbol{x}_\boldsymbol{v}_w)(\boldsymbol{x} - \boldsymbol{v}_w)^T - \boldsymbol{\Sigma}_w], \tag{9}$$

where α is defined as

$$\alpha = \lambda \cdot \exp(-100 f_w) \tag{10}$$

Here, λ is a constant having the meaning of the learning rate. For all the other rules in the population, except of the indefinite one, the prior probabilities f_i are modified as follows:

$$f_i \leftarrow (1 - \alpha) f_i. \tag{11}$$

3 Simulation

Feasibility of the proposed method is tested under simulation. The simulation setup is shown in Fig.2. The goal of robot is to avoid the wall and reach the light source. The robot has two light sensors a range of which is between 0 and 1. When the robot reach the goal, the light sensor strength is equivalent to 1. The

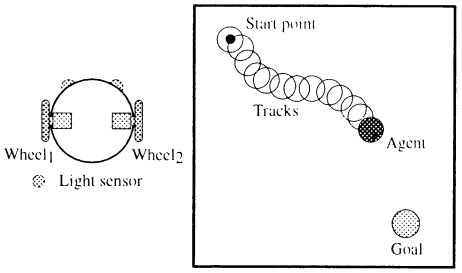
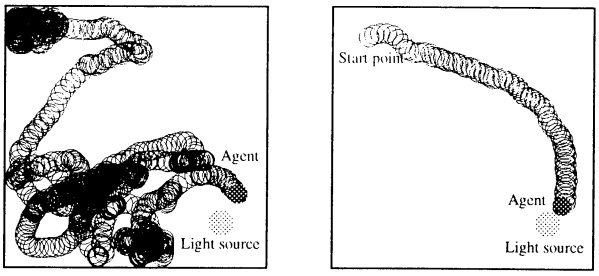


Figure 2: Simulation setup.

time interval between the starting motion and reaching the goal is called episode. The robot is set at the starting point, with its heading direction oriented randomly at every episode. The episodes are updated when the robot reaches the goal, or when the number of produced actions reaches 1000. The robot is rewarded only upon reaching the goal and is punished for every collision.

The direct payoff P=20 is set for rewarding the rules, and P=-5% of the corresponding utility is set for punishment. The maximum number of rules is restricted to n_r=100. When reproduction exceeds the limit, a definite rule with the minumum utility is removed. The parameters are set as follows: T_m=3.0, c_f=0.01, γ=0.8, η=0.98, κ=0.15, u_{min}=9.5, u_0=10, σ=0.005, λ=0.001.



(a) 1st episode. (b) 100th episode.

Figure 3: Acquired behavior

The robot behavior in episode No. 1 and No. 100 are shown in Fig.3. The number of collisions and the number of actions (step) are plotted in Fig.4. In the first episode the robot has no prior knowledge of the environment. Thus, the behavior of the robot is randomly and the number of collisions increase. After episode No. 20, the robot has acquired the light seeking, collision avoidance patterns. Fig.5 plots the num-

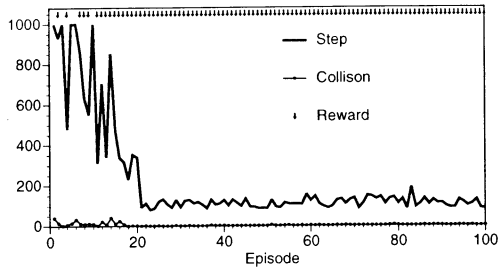


Figure 4: Learning history

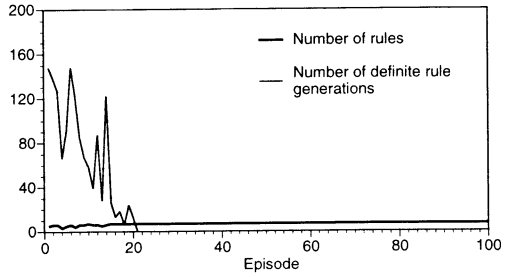
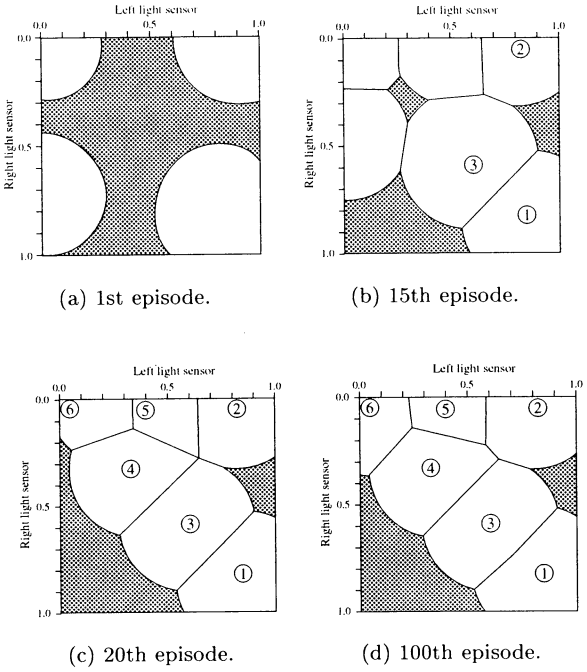


Figure 5: Rule production history



(a) 1st episode. (b) 15th episode.

(c) 20th episode. (d) 100th episode.

Figure 6: Segmented state space

ber of generations of the definite rules and the total number of rules in learning system. As can be seen, the number of generations of the definite rules increase during episode No. 20, and the total number of rules is stabilized at the level of seven. Fig.6 shows the segmented state space at each episode. The circle indicates the field covered with the definite rules, and the indefinite rule is covered by the gray area. As can be seen, the internal state space of the robot has been divided by seven clusters. Comparing Fig.6(c) with 6(d) indicates that the size of each cluster is modified very little as learning stabilizes.

4 Conclusions

A new reinforcement learning technique for adaptive segmentation of the continuous state space was formulated. The technique makes use of an unsupervised clustering method, where the segmented state space is represented by Bayes boundaries. The proposed technique has been applied to a navigation task and tested under simulation. The simulation results show that the robot can segment its continuous state space adaptively and acquires the behavior to reach the goal. In future, we plan to test the proposed technique under experiments with real Khepera robot.

Acknowledgements

This study has been conducted in "Methodology of Emergent Synthesis", Project (JSPS-RFTF96P00702) supported by the Japan Society for the Promotion of Science.

References

[1] L.P. Kaelbling, M.L. Littman, and A.W. Moore, Reinforcement Learning: A Survey, Journal of Artificial Intelligence Research, Vol. 4, 1996.

[2] K. Kuroyama, M.M. Svinin, Y. Nakamura, and K. Ueda, Learning control of autonomous robots with the use of an instance-based classifier generator in the continuous state space, Proc. of the 3rd International Symposium on Artificial Life and Robotics, AROB'98, Vol. 1, pp. 89-92, 1998.

[3] M.M. Svinin, K. Yamada, and K. Ueda, Reinforcement learning approach to acquisition of stable gaits for locomotion robots, Proc. of the IEEE International Conference on Systems, Man, and Cybernetics, vol. VI, pp. 936-941, 1999.

[4] T.Kohonen,Self-Organization and Associative Memory, Second Edition, Springer-Verlag, 1988.

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

The study on the optimization problem by neural network

Masanori Sugisaka and Naofumi Watanabe

Department of Electrical and Electronic Engineering, Oita University
700 Dannoharu, Oita 870-1192, JAPAN

msugi@cc.oita-u.ac.jp
naofumiw@cc.oita-u.ac.jp

Abstract

Recently, the research related to neuro-computer is increasing. In a word, a neuron is an information processing unit that has the same function in the brain. Even if used present brain science, we can not make clear the human brain function because it is complicated in structure and it has an ability of learning.

This research deals with learning of the neuron. In present, learning function of the neuron is quite different from the human because we are not able to adopt the real complicated learning process yet in the artificial neural network. In this research, we want to explain how the Hopfield network study combinatorial optimization problem.

The purpose of this study is to investigate the effects of parameters and weight coefficients under the some condition in order to achieve the high performance of learning.

We want to apply our results obtained for the study of the course recognition of mobile vehicle and so on in future.

1. Introduction

The neural network is different from digital computer that makes conventional processing having to do with logic triumphant and has the ability to learn. In other words, it has the ability to change itself as the fitting with the outside environment and to control it. Therefore, by giving the neural network learning data, it is possible to learn by it. As the neural network is used as a tool one have to consider, the pattern recognition and the memory system and so on.

We study how to take the data of the course choice using the optimization of the Hopfield model and how to learn about the optimization.

2. Neural network

The neural network is the one to have copied a cerebral nervous circuit network using the model of the artificial neuron. This neural network can be roughly divided to the

following two kinds by the way of the composition of each neuron.

One is the neural network of the mutual combination type of the combination model as it backs in the feeding in the output of the neuron itself, or each other the one of the neurons output and for the other neuron to input among two specific neurons. Another is a hierarchy-type neural network. There is not feed back loops among the neurons in this neural network and the signal merely spreads only through the one-sided direction only.

In this study time, we use a hierarchy-type neural network.

3. The optimization

There is the purpose when we will try the optimization. However, only to describe optimization in mathematically, the purpose must be set in mathematics. This is called a cost function. The cost function evaluates usefulness of this purpose. And if the usefulness is higher, value of cost function becomes smaller. Therefore, if we set the cost function, the optimization is paraphrased problem searched minimum of cost function.

However, a lot of problems included more than one purpose at the same time. In this case, it is necessary that cost function is expression of the total quantity of each purpose. But to trouble in this time, each problem is the case of relation of trade-off. At this time, to find the compromise points of demand disagreed is the optimization. However, even if we will try to find good compromise point, the position changes according to the importance of each purpose. Therefore, the cost function is expressed not, in a simple summation with weight incorporated the degree of importance for each function.

The optimization problem and its algorithm formulate a cost function, and it become minimum for a group of the variable.

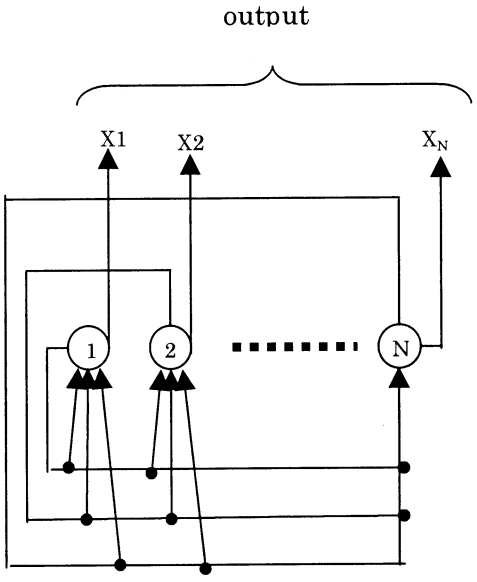


Fig.1 Hopfield-model

4. Simulation experiment

Conditions of optimization problem of combination are given as follows. First condition is "pass more optional spots at once and return to the first spot". Second condition is "the distance is the shortest". Because this problem has a lot of combination of route, we take much time in digital computer. Therefore, we solve this problem using artificial neural network.

We must set cost function to solve optimization problem. When we solve optimization problem using artificial neural

network, variable of cost function is output of each neural unit. Therefore, we must think how we express the turns of optional spots. For example, we think three spots, A_1, A_2 and A_3. Distance of each spot is known. At this time, nine units of 3×3 matrix is shown in Fig.2. Number of 1 or 0 are output of each neural unit. Output of 1 expresses to pass one spot, and output of 0 shows to pass other spots. As shown in Fig.2 the order is $A_2 \rightarrow A_1 \rightarrow A_3$.

The equation for the first condition is shown by Eq. (1).

$$L = \sum_{i=1}^{N} \sum_{K=1}^{N} \sum_{l \neq k} d_{kl} x_{ki} x_{l(i+1)} \qquad (1)$$

The equation for the 2nd condition is shown by Eq. (2).

$$C = \sum_{i}^{N} (\sum_{k}^{N} x_i - 1)^2 - \sum_{k}^{N} (\sum_{i}^{N} x_k - 1)^2 \quad (2)$$

Therefore, the cost function is given by Eq. (3).

$$F = \alpha L + \beta C \qquad (3)$$

(α and β are waits.)

	1	2	3
A_1	0	1	0
A_2	1	0	0
A_3	0	0	1

Fig.2 Example of matrix

5. Simulation result

In this study, we show a method how to solve optimization problem using Hopfield model.

In the first experiment, we stipulate that value of α is 3 (constant) and value of β is changed. α is a value connected with "minimum of distance". β is a value connected with "achieving the course of right choice". Therefore, value of β is made bigger and bigger, so the rate of choice of right course become higher and higher. However, this case is not best course because α become smaller. In the case of β =10, the distance is not minimum.

In the second experiment, value of β is fixed at constant from the result of the first experiment, and value of α is changed. In this experiment, value of β is fixed at 6 because the choice of course begins to succeed when the value of β becomes from 4 to 10. Value of α is made bigger and bigger, so minimum of distance becomes more and more important. However, the course doesn't become right. In the case of α=8, the course doesn't often achieve right.

In the last experiment, we used appropriate values such that "α=2 and β=6" or "α=3 and β=6" from result of experiments 1 and 2.

In this experiment, the course distance becomes small a little. If we increase the number of measurement and raise the performance, we will get the good result.

6. Conclusion

There are following two points for the study in future.

①The number of the parameters can be increased in order to use more data than the data of this time.

② Instead of the Hopfield model, Boltzmann machine network will be used also.

As the future problem, we want to apply proposed method for the study of the course recognition of mobile vehicle and so on.

References

[1] M.Sugisaka "The system theory and the control", in Japanese, Nikkankougyou Sinnbunnsya, pp.157～pp.160, 1997

[2] H. Itikawa "The hierarchy of type neural network", in Japanese, Kyouritsu Syuppann Kabusiki Gaisya, pp.1～pp.34, 1993

[3] T. Ibaragi, M. Fukushima "The technique of the optimization", in Japanese, Kyouritsu Syuppann Kabusiki Gaisya, pp.1～pp.3・pp.173～pp.196, 1993

[4] H. Haruhiko "The revision new C language guide", in Japanese, Softbank, pp.1～pp.34・pp.79～pp.122, 1991

[5] K/ Nakano "The information system of the brain to make with C", in Japanese, Kindai Kagakusya, pp.14～pp.26・pp.97～pp.124, 1995

Proc. of The Fifth Int. Symp. on Artificial Life and Robotics (AROB 5th'00),
Oita, Japan, 26-28, January, 2000,

The prediction amount of LD gas consumption by neural network

M. Sugisaka and H. Koga
Graduate School of Engineering, Department of Electrical and Electronic Engineering, Oita University
700 Danannoharu, Oita, 870-1192, JAPAN
Tel: +81-975-54-7831 Fax: +81-975-54-7820
E-mail: msugi@cc.oita-u.ac.jp hkoga@cc.oita-u.ac.jp

Abstract

In recent years, not only electric company but various industries produce electric power. Steel companies also produce electric power by using LD gas. First LD gas in the converter is sent to the gas holder and is saved in the gas holder. Second the gas is sent to the boiler to produce electric power. The process is repeated throughout the operation.

In this paper, we consider a method how to predict both amount of total LD gas produced and the time when LD gas is occurred by using neural network. Then we predict the amount of LD gas consumption by taking account of the amount of the LD gas occurred. The time when LD gas is occurred and level of the gas holder are also predicted using neural network.

1. Introduction

Ore is deoxidized and become metallic iron. The iron was made with the manganese, the silicon, the sulfur, the phosphorus and the carbon. The iron is called pig iron. This pig iron is refined more, removes impurity such as carbon with oxygen. The iron is called steel. Converter is furnace which removes impurity in the pig iron. Old converter (Bessemer converter and Thomas converter) adopts the way of making refine air in the oxidation by sending it from the bottom of furnace. In this way, the rate of nitrogen is high with nitrogen in air and the steel is fragile.

However, it blew only oxygen at the high pressure, the high speed from the top of the truth of pig iron and the way of refining was invented in Austria, 1953. Good quality steel be able to be produced.

2. Produce electric power by using LD gas

There are many ways to produce electric power, thermal power, hydraulic power and atomic power for intense. Steel companies produce electric power by using LD gas. Converter is operated in the process of steel making. Then a large quantity of LD gas is occurred in the converter. The gas in the converter is send to the gas holder. Next the gas is sent to the boiler to produce electric power.

We understand from Fig. 1 LD gas doesn't occur 15 minutes if gas occur 15 minutes. In Fig.1 the attenuate line is amount of gas occurred, the thick line is gas consumption, the dotted line is level of gas holder. Amount of gas consumption is constant. Level of gas holder depend on amount of gas occurred and amount of gas consumption.

To increase LD gas in the gas holder is go up level of the gas holder is. To decrease LD gas in the gas holder is to go down level of the gas holder is low. In this paper control the level of the gas holder is controlled by using neural network.

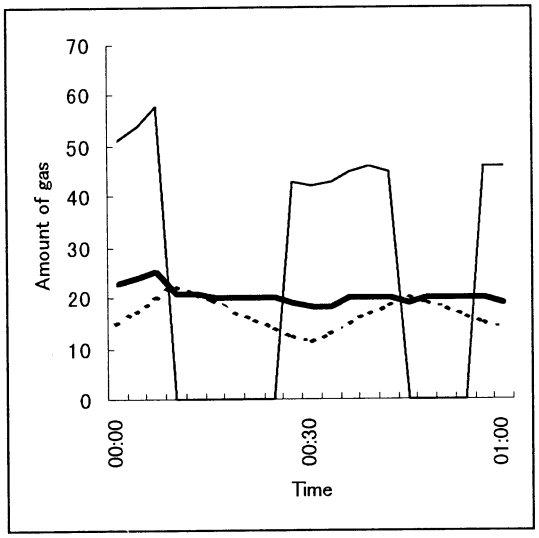


Fig. 1 LD gas versus time

3. The propose method

3.1 neural network

In this paper control the level of the gas holder by using neural network. The movement of the neuron of neural network is controlled by the equation such that

$$dU_i/dt = -\partial E(V_1, V_2, \ldots V_n)/\partial V_i$$

The neural network changes Ui which shows the state of neurons. The function Ui is given into the output function and decides the firing situation of each neuron specified by V. The change of input Ui will respect to time, namely, dUi/dt is determined by considering the effect to the total energy E(V1,V2,...Vn) when changing the value. The reason why the right side mark is negative is because the one with the bigger firing situation Vi obtains a low value with stable energy E and approaches a solution.Neural network repeats a series of this work many times and solves an optimizing problem.

3.2 The composition of the system

First, we predict amount of total LD gas occurred in the convention.
Amounts of hot metal, amount of Scrap, amount of Oxygen, amount of Sulfur and time of refining are given into the input layer.
We composes a system to get amount of the gas in the output layer as shown in Fig.2.

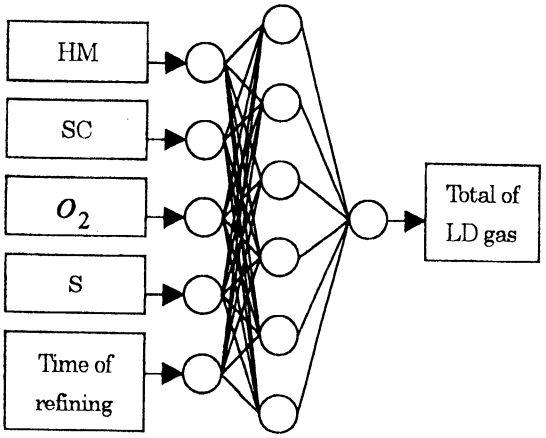


Fig.2 Prediction of amount of gas by neural network

Next, we predict time of the occurring gas by using neural network. In Fig.3 we show neural network to estimate the time occurring gas. The inputs of this neural are the times occurring gas in past a, b and c. The neural network system is used to estimates the current time occurring gas x as shown in Fig.3.

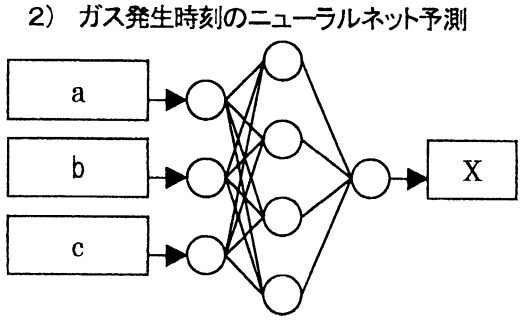


Fig.3 Time of occurring gas by neural network

Finally, amount of occurring gas, time of occurring gas and level of gas holder are given into the input layer. We composes a system to obtain optimal distribution amount of LD gas consumption in the output layer as shown in Fig.4.

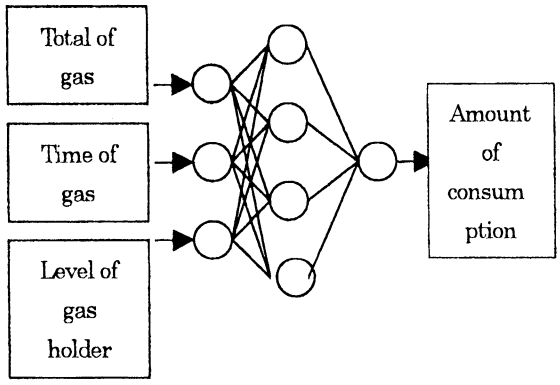


Fig.4 Prediction amount of LD gas consumption by neural network

4. Conclusion

We predicted optimal distribution amount of LD gas consumption. In this paper, we apply neural network. In the fact, There are various situations that amount of gas occurred increases rapidly or decreases rapidly and LD gas consumption is very large or very small. There is necessity to make theory is able to endure the problem.

Reference

[1] A. Ito, "How Do selfish Agents Learn to Cooperate?", Proceeding of 5th International Conference on Artificial Life, pp142-149, 1996.

[2] K. Hirasawa, M. Obayashi and J. Murata, "Universal Learning Network and Computation of its Higher Order Derivatives", Proceeding of 1995 IEEE International Conference on Neural Networks, pp1273-1277, 1995.

[3] K. Hirasawa, M. Obayashi M.Koga and M.harada, "Forward Pro-pagation Universal Learning Network", Proceeding of 1996 IEEE Inter-national Conference on Neural Networks, pp353-358, 1995.

[4] M. Sugisaka, Z. J. Liu, "Artificial Neural Network Models using Neurocomputer" Proceeding of the 2nd Int. Workshop on Advanced Mechatronics, Nagasaki, Japan, pp80-83, 1997.

[5] M. Sugisaka, Z. J. Liu, "The Application of Neurocomputer for a Control Problem", Proceeding of the 3nd Int. Symp. on Artificial Life and Robotics, Beppu, Oita, pp444-487, 1998.

[6] M. Sugisaka, Z. J. Liu, "The Simulation of Artificial Nneural Network Methods in Cooler Control using Neurocomputer for a Control Problem", Proceeding of the 3nd Workshop of Int., Institute for General Systems Studies,Qinhuandgdao,China, pp, 140-143, 1998.

[7] The Japanese steel society, "A Way of making pig iron and steel", pp182-220.

[8] F. Toussaint, H. Kuromura, "How to produce steel", pp 57-73.

Author Index

[A]

Abe, K. 268, 281, 484, 721, 725
Adachi, T. 597
Ahson, S. I. 342
Aihara, K. 157, 161, 165, 169, 173, 177, 185
Aito, H. 647
Akimoto, K. 717
Akiyama, Y. 729
Akutsu, T. 733
Alarcon, E. 821
Amari, S. P-2
Ando, H. 410
Ando, M. 729
Aoki, T. 424
Aota, Y. 838
Aoyama, T. 268, 539, 543, 825
Arif, M. 711
Arita, M. 737
Arita, T. 260
Asano, F. 503

[B]

Baba, T. 84
Bae, D. S. 833
Bae, J. I. 617, 833
Bahk, W. M. 5
Bamba, E. 495
Bedau, M. A. P-3
Berry, R. 19
Billinghurst, M. 22
Botelho, F. 846
Bubnicki, Z. I-1
Buller, A. 192, 204

[C]

Cabestany, J. 821
Cao, G. 228
Casti, J. L. B-1
Chae, J. H. 5
Chang, Y. H. 272
Chaudron, L. 641
Chen, X. 681
Cho, S. B. 334, 406
Chodakowski, T. 192
Choi, J. W. 272
Choi, Y. 402
Choi, Y. K. 116, 480
Chonan, S. 50
Chongstitvatana, P. 251
Chu, G. W. 317
Chun, H. B. 338
Chung, M. J. 313, 317, 375
Chung, T. J. 62
Chung, W. K. 402
Collins, J. J. 112, 247, 305, 347, 351
Cosp, J. 821

[D]

Dai, F. 568
Dai, R. I-16
D'Angelo, A. 363
Dassanayake, P. 773
Ding, X. 66
Doya, K. 255

[E]

Eaton, M. 112, 305, 347
Eggenberger, P. 208
Eom, T. D. 387, 391

[F]

Floreano, J. D. I-30
Fujii, M. 416
Fujiki, H. 563
Fujinaka, T. 665, 769
Fujisawa, K. 424

Fujiwara, Y. 450
Fukuda, T. 309
Funami, Y. 289, 625
Funamori, M. 141
Furuhashi, T. 88, 416
Furuta, K. 503

[G]
Ge, S. 805
Gomes, M. R. 264
Goto, S. 383

[H]
Hajiri, K. 458
Han, F. 677
Han, S. H. 272
Hanazaki, I. 462
Haneda, H. 104, 609
Hara, K. 488
Hara, M. 593
Harashima, F. P-1
Haseda, T. 613
Hashida, M. 801
Hashida, T. 66
Hashidume, K. 515
Hashimoto, H, 70, 80, 84
Hashimoto, T. 76
Hattori, Y. 88
Hayakawa, S. 424
Hayakawa, W. 643
Hayashi, H. 169
Hemmi, H. 192, 196
Hernandez, G. 846
Higashi, T. 309
Hikage, T. 196
Hirakawa, M. 534
Hirasawa, K. 753, 842
Hirayama, H. 629, 637
Hirayama, K. 80
Hisano, M. 54
Holland, O. 470
Hong, D. P. 62
Honma, N. 281
Honma, S. 813
Hori, G. 185
Horiguchi, Y. 689
Horio, Y. 169
Hu, D. 108
Hu, J. 753
Hur, H. R. 609
Hwang, S. Y. 62

[I]
Ichiki, T. 137
Ide, J. 293
Igarashi, H. 297
Igarashi. T. 232
Ikeguchi, T. 181
Inabayashi, S. 547, 643
Inamoto, T. 809
Inooka, H. 711
Inou, N. 511
Inoue, H. 454
Ioi, K. 297
Ishibashi, M. 534
Ishido, M. 200
Ishihara, T. 711
Ishii, H. 693
Ishii, S. 255, 661
Ishikawa, K. 371
Ishitani, N. 92
Itakura, H. 817
Ito, K. 693
Ito, M. 255
Itou, K. 589
Itou, M. 216
Itou, O. 657
Iwamoto, T. 633
Izumi, K. 669, 673, 677, 681, 773

[J]
Jang, H. M. 480
Jeong, J. 5
Jeong, S. W. 829
Jiang, R. 108
Jiang, Z. 50
Jin, C. 753
Jo, I. H. 379
Jo, Y. G. 559

Johnson, J. I-20
Jones, S. 26
Joung, J. G. 321
Ju, D. Y. 613
Jun, T. Y. 5
Jung, J. H. 617

[K]
Kamaya, H. 484
Kamimura, R. 277
Kanbe, D. 572
Kang, H. 329, 379, 528, 559
Kang, H. Y. 62
Katada, Y. 580
Katai, O. 434, 438, 446, 454, 685
Kato, H. 22
Katoh, R. 367
Katou, Y. 643
Kawai, N. 34, 40, 45
Kawai, T. 410
Kawakami, H. 685
Kawakita, M. 238
Kawamura, Y. 793
Kawano, S. 805
Kawano, Y. 703
Kazui, T. 629, 637
Kelly, I. 470
Kiguchi, K. 669, 673, 677, 681, 773
Kim, D. H. 375
Kim, D. J. 5
Kim, D. J. 480
Kim, H. S. 1
Kim, J. H. 375
Kim, K. S. 5
Kim, S. 116, 480
Kim, S. I. 149
Kim, S. K. 325
Kim, S. Y. 5
Kim, T. K. 313
Kim, Y. B. 480
Kinone, T. 92
Kinouchi, Y. 488, 547
Kitagawa, N. 629, 637
Kitamura, S. 809
Kitazoe, T. 137, 141, 149, 153
Kobayashi, I. 173
Koga, H. 858
Kohata, N. 84
Kondo, H. 523
Kong, S. G. 325, 355
Kryssanov, V. V. 809
Ku, Y. S. 62
Kuge, M. 613
Kuhara, S. 733
Kurematsu, Y. 809
Kuribayashi, K. 54, 58, 66
Kurimoto, T. 224
Kushida, D. 383
Kushida, S. 613
Kyura, N. 383

[L]
Lee, B. R. 829
Lee, C. Y. 387
Lee, D. J. 116
Lee, D. W. 338
Lee, H. 484
Lee, J. H. 70
Lee, J. H. 395
Lee, J. J. 387, 391
Lee, J. M. 104, 272, 609, 833
Lee, K. J. 321
Lee, M. H. 104, 272, 617
Lee, M. J. 116
Lee, M. C. 272
Lee, W. H. 528
Lee, Y. J. 313
Liu, J. Q. 777, 781, 785
Liu, Z. 212
Lu, B. 842
Luo, Y. 108

[M]
Madrenas, J. 821
Maekawa, T. 34, 40, 45
Maeshiro, T. 450
Maille, N. 641
Makino, Y. 717

Mansour, A. 359
Matsumoto, N. 547
Matsumura, Y. 580
Matsunaga, N. 801
Matsuno, F. 507
Matsuzaki, M. 410
May, R. 22
Melhuish, C. 470
Mignonneau, L. 124
Mishiro, S. 491
Mita, T. 499
Mitsuo, N. 601
Miyake, Y. 838
Miyano, S. 733
Mizuhara, H. 801
Mizumoto, M. 420
Mizuno, Y. 185
Mizutani, M. 547
Mogi, K. 507
Monma, M. 232
Montesello, F. 363
Moon, S. W. 355
Moreno, J. M. 821
Morimoto, T. 450
Mukaidono, M. 96
Murakami, K. 54
Murakami, T. 534
Murao, H. 809
Murata, J. 753, 842

[N]

Nagayuki, Y. 255
Nakagawa, M. 519
Nakamine, H. 699
Nakamura, A. 466
Nakamura, M. 293, 383
Nakata, A. 420
Nakatsu, R. 13
Nakazato, K. 495
Namiki, Y. 50
Nanayakkara, T. 669
Nawa, N. E. 446
Naya, F. 789
Negoro, K. 801
Nian, W. 132
Nijo, T. 576
Nino, F. 846
Ninomiya, Y. 476
Nishimura, H. 153
Nishina, E. 34, 40, 45
Nishinomiya, N. 651
Nitta, H. 80
Nitta, S. 721
Nobuki, O. 797
Noguchi, T. 729
Numata, M. 268

[O]

Ogasawara, J. 801
Ogasawara, T. 466
Ogata, T. 793
Ogishima, S. 741, 745
Oh, S. R. 402
Ohkura, K. 580
Ohnishi, K. 572
Ohtani, S. 200
Oka, M. 757
Okada, H. 657
Okada, M. 434, 438, 454
Okada, N. 438
Okamoto, M. 534
Okazaki, K. 132
Oki, T. 765
Okita, Y. 629, 637
Okuhara, K. 555
Okuma, M. 515
Okuma, S. 424
Okuma, Y. 185
Okumura, K. 511
Omatu, S. 665, 761, 769
Omori, T. 657
Onat, A. 703
O'Neill, M. 247, 347, 351
Onizuka, K. 729
Ono, I. 563, 576
Ono, N. 563, 576
Oohashi, T. 34, 40, 45
Oura, K. 462
Ozaki, T. 189

[P]

Pagello, E. 363
Park, J. H. 609
Park, J. W. 609
Park, K. Y. 829
Park, S. T. 617
Poupyrev, I. 22
Poveda, A. 821
Prautsch, P. 499

[Q]

Quas, A. 846

[R]

Ray, T. S. I-12
Reimann, S. 359
Ren, F. 741, 745
Ryan, C. 351

[S]

Sagara, S. 367
Sakai, M. 281
Sakai, Y. 797
Sakamoto, K. 621
Sakamoto, T. 54
Sakuma, H. 515
Sakumura, Y. 161
Sakurai, A. 371
Sannomiya, N. 699
Sano, M. 725
Sapaty, P. S. 301
Sato, C. 462
Sato, M. 717
Satoh, T. 757
Sawaragi, T. 434, 689
Seki, S. 605
Sekiguchi, M. 120
Sekiyama, K. 309
Seok, H. S. 321
Sepulveda, T. 264
Serikawa, S. 128
Shah, S. L. 765
Sheehan, L. 112, 305, 347
Shi, Z. 189
Shibasaki, H. 293
Shibata, K. 589
Shimada, T. 410
Shimoda, H. 693
Shimohara, K. 34, 40, 45, 192, 196, 200, 204, 255, 430, 438, 446, 454, 458, 777, 781, 785
Shimomura, T. 128
Shin, S. 238
Shinchi, T. 153
Shinozawa, K. 789
Shiose, T. 434
Shono, M. 220
Sim, K. B. 338, 379
Sommerer, C. 124
Son, K. 272
Song, G. B. 406
Suehiro, T. 466
Sugawara, K. 725
Sugi, T. 293
Sugihara, K. 145
Sugisaka, M. I-7, 212, 216, 220, 224, 399, 476, 491, 568, 584, 593, 597, 601, 605, 621, 647, 854, 858
Sun, J. 551
Suzuki, H. 30
Suzuki, R. 260
Suzuki, T. 424
Suzuki, Y. 749
Svinin, M. 580, 850

[T]

Tabuse, M. 145, 153
Tagawa, K. 104, 609
Takadama, K. 430, 438, 446, 454, 458
Takagi, T. 80, 96
Takeda, H. 721
Takeuchi, Y. 54
Tamakoshi, H. 661
Tamura, M. 367
Tamura, S. 132, 643
Tamura, Y. 189

Tanaka, A. 721
Tanaka, H. 165
Tanaka, H. 728, 741, 745, 749
Tanaka, K. 757
Tanaka, T. 555
Tang, J. 551
Tanikawa, T. 367
Tano, S. 92
Tarumoto, T. 58
Taylor, C. E. I-6
Teal, T. I-6
Teshima, T. 539, 543, 825
Toba, O. 685
Tokuda, I. 177
Tongchim, S. 251
Tosa, N. 9
Tsuchiya, K. 703
Tsujita, K. 703
Tsukune, H. 466

[U]
Uchibori, A. 757
Udawatta, L. 673
Ueda, K. 580, 850
Ueda, M. 589
Ueno, O. 34, 40, 45
Ujihashi, S. 511
Ukai, S. 838
Urzelai, J. I-30

[W]
Wakamatsu, H. 813
Wang, X. 584
Wang, X. 399
Watanabe, K. 669, 673, 677, 681, 773
Watanabe, M. 157
Watanabe, N. 854
Watanabe, T. 725
Wu, H. 242
Wu, J. L. 797, 801, 805
Wu, W. 693

[X]
Xiong, S. 242
Xu, C. I-12

[Y]
Yamada, K. 285, 289, 625
Yamada, K. 850
Yamaguchi, S. 817
Yamaguchi, T. 70
Yamaguchi, T. 80, 84, 96
Yamaguchi, T. 442
Yamakawa, H. 657
Yamakita, M. 503
Yamamoto, G. 488
Yamamoto, H. 813
Yamamoto, T. 765
Yamasaki, K. 120, 488
Yamato, J. 789
Yamauchi, S. 801
Yamawaki, M. T. 532
Yambe, T. 721
Yang, S. Y. 617
Yano, M. 717
Yasunaga, M. 268
Yokouchi, T. 643
Yoneda, M. 173
Yonezawa, Y. 232, 651
Yoo, D. H. 317
Yoshida, A. 100
Yoshida, S. 532
Yoshihara, I. 268, 539, 543, 725, 825
Yoshikawa, H. 693
Yoshioka, M. 761, 769
Yoshitomi, Y. 149
Yoshizawa, M. 721
Yum, M. K. 1

[Z]
Zhang, B. T. 321
Zhang, G. 242
Zhang, L. 523
Zhang, X. 813
Zhang, Y. 228
Zhang, Y. G. I-7, 551
Zhou, D. I-16
Zhou, Z. 242
Zhu, H. 539, 543, 825
Zhu, X. 228